H3C SecPath M9000 系列 多业务安全网关

虚拟化技术配置指导(V7)

新华三技术有限公司
http://www.h3c.com

前 言

本配置指导主要介绍虚拟化技术的相关功能原理及配置。

前言部分包含如下内容：

- 读者对象
- 本书约定
- 资料意见反馈

读者对象

本手册主要适用于如下工程师：

- 网络规划人员
- 现场技术支持与维护人员
- 负责网络配置和维护的网络管理员

本书约定

1. 命令行格式约定

格　式	意　义
粗体	命令行关键字（命令中保持不变、必须照输的部分）采用**加粗**字体表示。
斜体	命令行参数（命令中必须由实际值进行替代的部分）采用*斜体*表示。
[ ]	表示用“[ ]”括起来的部分在命令配置时是可选的。
{ x | y | … }	表示从多个选项中仅选取一个。
[x | y | …]	表示从多个选项中选取一个或者不选。
{ x | y | … }*	表示从多个选项中至少选取一个。
[x | y | …]*	表示从多个选项中选取一个、多个或者不选。
&<1-n>	表示符号&前面的参数可以重复输入1～n次。
#	由“#”号开始的行表示为注释行。

2. 图形界面格式约定

格　式	意　义
< >	带尖括号“< >”表示按钮名，如“单击<确定>按钮”。
[ ]	带方括号“[ ]”表示窗口名、菜单名和数据表，如“弹出[新建用户]窗口”。
/	多级菜单用“/”隔开。如[文件/新建/文件夹]多级菜单表示[文件]菜单下的[新建]子菜单下的[文件夹]菜单项。

3. 各类标志

本书还采用各种醒目标志来表示在操作过程中应该特别注意的地方，这些标志的意义如下：

警告	该标志后的注释需给予格外关注，不当的操作可能会对人身造成伤害。
注意	提醒操作中应注意的事项，不当的操作可能会导致数据丢失或者设备损坏。
提示	为确保设备配置成功或者正常工作而需要特别关注的操作或信息。
说明	对操作内容的描述进行必要的补充和说明。
窍门	配置、操作、或使用设备的技巧、小窍门。

4. 图标约定

本书使用的图标及其含义如下：

	该图标及其相关描述文字代表一般网络设备，如路由器、交换机、防火墙等。
	该图标及其相关描述文字代表一般意义下的路由器，以及其他运行了路由协议的设备。
	该图标及其相关描述文字代表二、三层以太网交换机，以及运行了二层协议的设备。
	该图标及其相关描述文字代表无线控制器、无线控制器业务板和有线无线一体化交换机的无线控制引擎设备。
	该图标及其相关描述文字代表无线接入点设备。
	该图标及其相关描述文字代表无线终结单元。
	该图标及其相关描述文字代表无线终结者。
	该图标及其相关描述文字代表无线Mesh设备。
	该图标代表发散的无线射频信号。
	该图标代表点到点的无线射频信号。
	该图标及其相关描述文字代表防火墙、UTM、多业务安全网关、负载均衡等安全设备。
	该图标及其相关描述文字代表防火墙插卡、负载均衡插卡、NetStream插卡、SSL VPN插卡、IPS插卡、ACG插卡等安全插卡。

5. 示例约定

由于设备型号不同、配置不同、版本升级等原因，可能造成本手册中的内容与用户使用的设备显示信息不一致。实际使用中请以设备显示的内容为准。

本手册中出现的端口编号仅作示例，并不代表设备上实际具有此编号的端口，实际使用中请以设备上存在的端口编号为准。

资料意见反馈

如果您在使用过程中发现产品资料的任何问题，可以通过以下方式反馈：

E-mail：info@h3c.com

感谢您的反馈，让我们做得更好！

目 录

1 IRF ····· 1-1
1.1 IRF 简介 ····· 1-1
1.1.1 IRF 组网示意图 ····· 1-1
1.1.2 IRF 的优点 ····· 1-2
1.1.3 IRF 基本概念 ····· 1-3
1.1.4 IRF 的连接拓扑 ····· 1-5
1.1.5 角色选举 ····· 1-6
1.1.6 IRF 中的接口命名规则 ····· 1-6
1.1.7 IRF 中的文件系统命名规则 ····· 1-7
1.1.8 IRF 中的配置同步 ····· 1-7
1.1.9 MAD 功能 ····· 1-8
1.1.10 MAD 检测机制 ····· 1-10
1.2 IRF 与硬件适配关系 ····· 1-15
1.3 IRF 配置限制和指导 ····· 1-16
1.3.1 硬件兼容性相关配置限制和指导 ····· 1-16
1.3.2 软件版本要求 ····· 1-16
1.3.3 IRF 规模 ····· 1-16
1.3.4 确定 IRF 物理端口 ····· 1-16
1.3.5 选择连接 IRF 端口的模块 ····· 1-17
1.3.6 IRF 物理端口连接要求 ····· 1-17
1.3.7 IRF 物理端口配置限制和指导 ····· 1-17
1.3.8 IRF 与其它软件特性的兼容性与限制 ····· 1-18
1.3.9 IRF 中 License 安装一致性要求 ····· 1-18
1.3.10 配置回滚限制 ····· 1-18
1.4 IRF 配置任务简介 ····· 1-18
1.5 配置准备 ····· 1-19
1.6 搭建 IRF ····· 1-20
1.6.1 配置成员编号 ····· 1-20
1.6.2 配置成员优先级 ····· 1-20
1.6.3 配置 IRF 端口 ····· 1-20
1.6.4 将当前配置保存到设备的下次启动配置文件 ····· 1-21
1.6.5 连接 IRF 物理接口 ····· 1-21
1.6.6 切换到 IRF 模式 ····· 1-21

1.6.7 访问 IRF ····· 1-22
1.7 配置 MAD ····· 1-22
1.7.1 配置限制和指导 ····· 1-22
1.7.2 配置 LACP MAD 检测 ····· 1-22
1.7.3 配置 BFD MAD 检测 ····· 1-23
1.7.4 配置 ARP MAD 检测 ····· 1-25
1.7.5 配置 ND MAD 检测 ····· 1-26
1.7.6 配置保留接口 ····· 1-28
1.7.7 MAD 故障恢复 ····· 1-28
1.8 调整和优化 IRF ····· 1-29
1.8.1 配置成员编号 ····· 1-29
1.8.2 配置成员优先级 ····· 1-29
1.8.3 配置 IRF 端口 ····· 1-30
1.8.4 快速配置 IRF 基本参数 ····· 1-31
1.8.5 开启 IRF 合并自动重启功能 ····· 1-32
1.8.6 配置成员设备的描述信息 ····· 1-32
1.8.7 配置 IRF 链路的负载分担模式 ····· 1-33
1.8.8 配置 IRF 的桥 MAC 地址 ····· 1-34
1.8.9 开启启动文件的自动加载功能 ····· 1-35
1.8.10 配置 IRF 链路 down 延迟上报功能 ····· 1-36
1.8.11 拆卸 IRF 物理端口所在的接口模块扩展卡 ····· 1-36
1.8.12 更换 IRF 物理端口所在的接口模块扩展卡 ····· 1-36
1.9 IRF 显示和维护 ····· 1-36

1 IRF

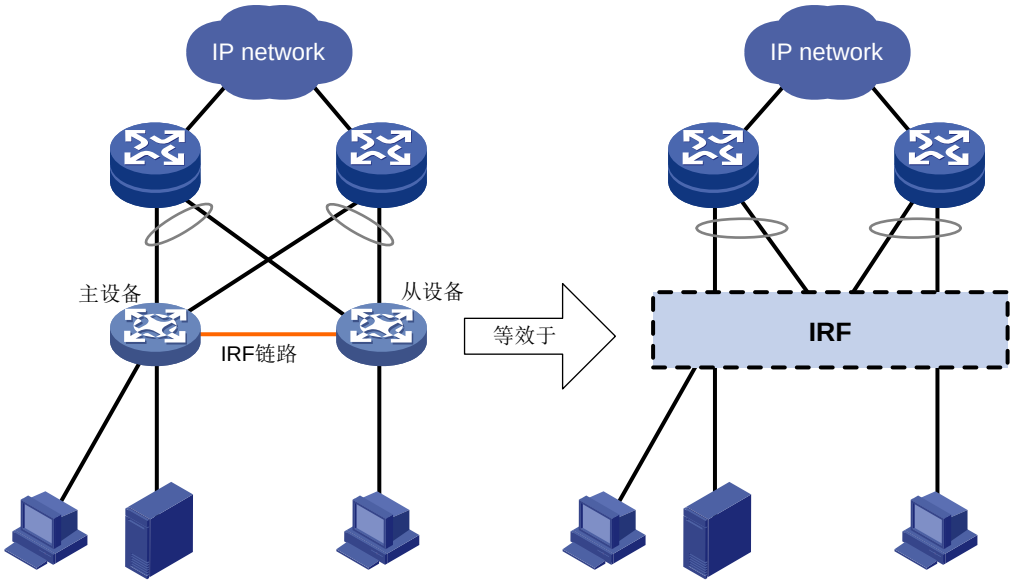
1.1 IRF简介

IRF（Intelligent Resilient Framework，智能弹性架构）是 H3C 自主研发的软件虚拟化技术。它的核心思想是将多台设备连接在一起，进行必要的配置后，虚拟化成一台设备。使用这种虚拟化技术可以集合多台设备的硬件资源和软件处理能力，实现多台设备的协同工作、统一管理和不间断维护。为了便于描述，这个“虚拟设备”也称为 IRF。所以，本文中的 IRF 有两层意思，一个是指 IRF 技术，一个是指 IRF 设备。

1.1.1 IRF 组网示意图

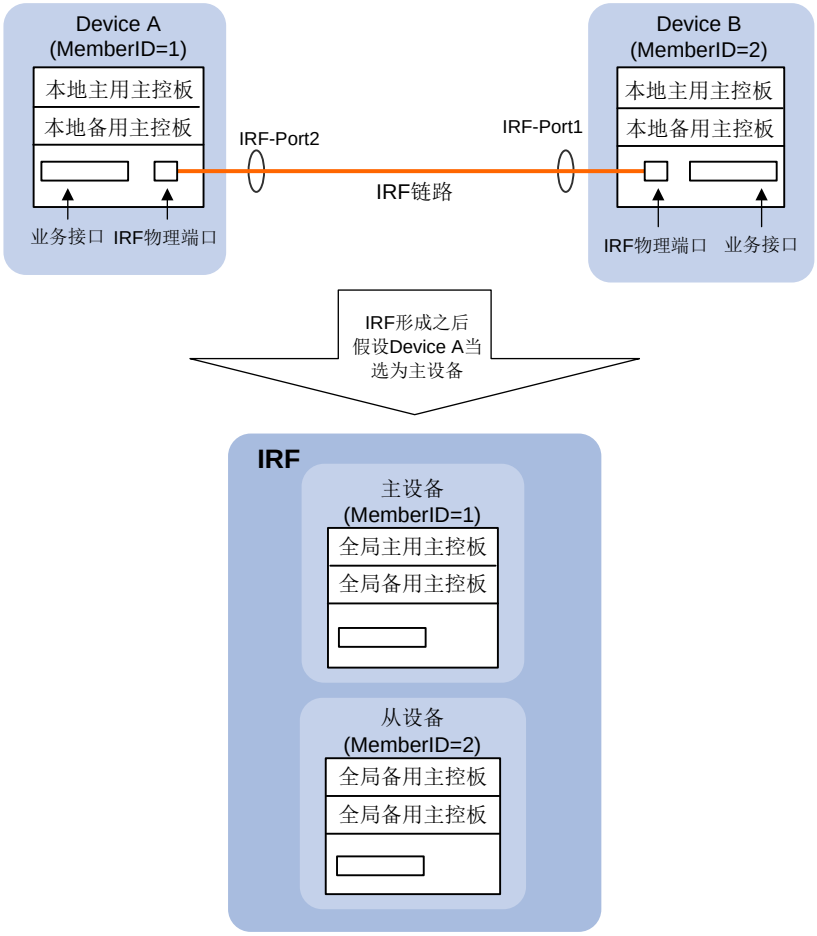
如图 1-1 所示，两台设备组成 IRF，对上、下层设备来说，它们就是一台设备——IRF。所有成员设备上的资源归该虚拟设备 IRF 拥有并由主设备统一管理。

图1-1 IRF 组网应用示意图



如图 1-2 所示，Device A 和 Device B 组成 IRF 后，IRF 拥有四块主控板（一块主用主控板，三块备用主控板），两块接口板。IRF 统一管理 Device A 和 Device B 的物理资源和软件资源。

图1-2 IRF 虚拟化示意图



1.1.2 IRF 的优点

IRF 主要具有以下优点：

- 简化管理：IRF 形成之后，用户通过任意成员设备的任意端口都可以登录 IRF 系统，对 IRF 内所有成员设备进行统一管理。
- 1:N 备份：IRF 由多台成员设备组成，其中，主设备负责 IRF 的运行、管理和维护，从设备在作为备份的同时也可以处理业务。一旦主设备故障，系统会迅速自动选举新的主设备，以保证业务不中断，从而实现了设备的 1:N 备份。
- 跨成员设备的链路聚合：IRF 和上、下层设备之间的物理链路支持聚合功能，并且不同成员设备上的物理链路可以聚合成一个逻辑链路，多条物理链路之间可以互为备份也可以进行负载分担，当某个成员设备离开 IRF，其它成员设备上的链路仍能收发报文，从而提高了聚合链路的可靠性。
- 强大的网络扩展能力：通过增加成员设备，可以轻松自如地扩展 IRF 的端口数、带宽。因为各成员设备都有 CPU，能够独立处理协议报文、进行报文转发，所以 IRF 还能轻松自如的扩展处理能力。

1.1.3 IRF 基本概念

1. 运行模式

设备支持两种运行模式：

- 独立运行模式：处于该模式下的设备只能单机运行，不能与别的设备形成 IRF。
- IRF 模式：处于该模式下的设备可以与其它设备互连形成 IRF。

2. 成员设备的角色

IRF 中每台设备都称为成员设备。成员设备按照功能不同，分为两种角色：

- 主用设备（简称为主设备）：负责管理和控制整个 IRF。
- 从属设备（简称为从设备）：处理业务、转发报文的同时作为主设备的备份设备运行。当主设备故障时，系统会自动从从设备中选举一个新的主设备接替原主设备工作。

主设备和从设备均由角色选举产生。一个 IRF 中同时只能存在一台主设备，其它成员设备都是从设备。关于设备角色选举过程的详细介绍请参见“1.1.5 角色选举”。

3. 成员设备编号

IRF 使用成员设备编号来标识和管理成员设备。接口名称和文件系统路径中均包含成员设备编号，以此来唯一标识 IRF 设备上的接口和文件。

每台成员设备必须具有唯一的编号。如果两台设备的成员编号相同，则不能组成 IRF。如果新设备加入 IRF，但是该设备的成员编号与已有成员设备的编号冲突，则该设备不能加入 IRF。

4. 主控板的角色

设备加入 IRF 后，设备上的主控板就具有两重身份（身份不同责任不同）：

- 本地身份：负责管理本设备的事宜，比如主用主控板和备用主控板间的同步、协议报文的处理、路由表项的生成维护等。
- 全局身份：负责处理 IRF 相关事宜，比如角色选举、拓扑收集等。

表1-1 主控板的角色

主控板角色	描述
本地主用主控板	成员设备的主用主控板，负责管理本台设备，是成员设备的必备硬件
本地备用主控板	成员设备的备用主控板，是本地主用主控板的备份，是成员设备的可选硬件
全局主用主控板	IRF的主用主控板，负责管理整个IRF，就是主设备的本地主用主控板
全局备用主控板	IRF的备用主控板，是全局主用主控板的备份。除了全局主用主控板，IRF中所有成员设备的主控板均为全局备用主控板

5. 成员优先级

成员优先级是成员设备的一个属性，主要用于角色选举过程中确定成员设备的角色。优先级越高当选为主设备的可能性越大。

设备的缺省优先级均为 1，如果想让某台设备当选为主设备，则在组建 IRF 前，可以通过命令行手工提高该设备的成员优先级。

6. IRF 端口

一种专用于 IRF 成员设备之间进行连接的逻辑接口，每台成员设备上可以配置两个 IRF 端口，分别为 IRF-Port1 和 IRF-Port2。它需要和物理端口绑定之后才能生效。

（支持模式切换的设备）IRF 端口编号规格如下：

- 在独立运行模式下，IRF 端口采用一维编号，编号为 IRF-Port1 和 IRF-Port2；
- 在 IRF 模式下，IRF 端口采用二维编号，编号为 IRF-Port*n*/1 和 IRF-Port*n*/2，其中 *n* 为设备的成员编号。

为简洁起见，本文描述时统一使用 IRF-Port1 和 IRF-Port2。

IRF 端口的状态由与它绑定的 IRF 物理端口的状态决定。与 IRF 端口绑定的所有 IRF 物理端口状态均为 down 时，IRF 端口的状态才会变成 down。

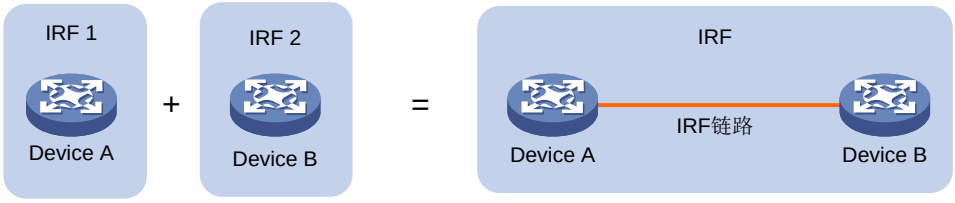
7. IRF 物理端口

与 IRF 端口绑定，用于 IRF 成员设备之间进行连接的物理接口。IRF 物理端口负责在成员设备之间转发 IRF 协议报文以及需要跨成员设备转发的业务报文。

8. IRF 合并

如图 1-3 所示，两个（或多个）IRF 各自已经稳定运行，通过物理连接和必要的配置，形成一个 IRF，这个过程称为 IRF 合并。

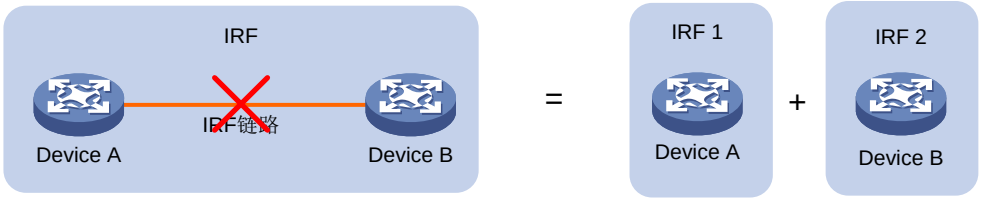
图1-3 IRF 合并示意图



9. IRF 分裂

如图 1-4 所示，一个 IRF 形成后，由于 IRF 链路故障，导致 IRF 中两相邻成员设备不连通，一个 IRF 分裂成两个 IRF，这个过程称为 IRF 分裂。

图1-4 IRF 分裂示意图



10. MAD

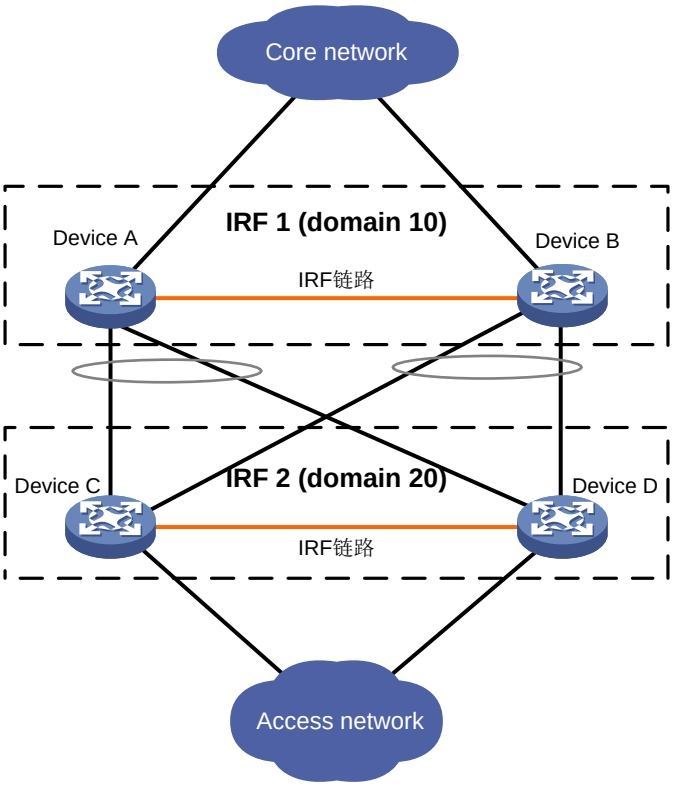
IRF 链路故障会导致一个 IRF 分裂成多个新的 IRF。这些 IRF 拥有相同的 IP 地址等三层配置，会引起地址冲突，导致故障在网络中扩大。MAD（Multi-Active Detection，多 Active 检测）机制用来进行 IRF 分裂检测、冲突处理和故障恢复，从而提高系统的可用性。

11. IRF 域

域是一个逻辑概念，一个 IRF 对应一个 IRF 域。

为了适应各种组网应用，同一个网络里可以部署多个 IRF，IRF 之间使用域编号（DomainID）来区别。如图 1-5 所示，Device A 和 Device B 组成 IRF 1，Device C 和 Device D 组成 IRF 2。如果 IRF 1 和 IRF 2 之间有 MAD 检测链路，则两个 IRF 各自的成员设备间发送的 MAD 检测报文会被另外的 IRF 接收到，从而对两个 IRF 的 MAD 检测造成影响。这种情况下，需要给两个 IRF 配置不同的域编号，以保证两个 IRF 互不干扰。

图1-5 多 IRF 域示意图

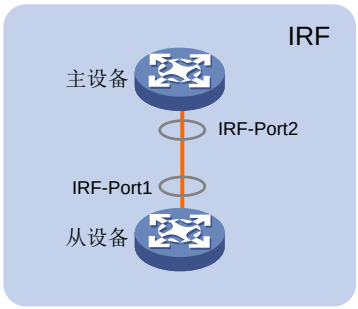


1.1.4 IRF 的连接拓扑

IRF 的连接拓扑为链形连接，如图 1-6 所示。

链形连接对成员设备的物理位置要求低，主要用于成员设备物理位置分散的组网。成员设备之间不允许连接中继设备

图1-6 IRF 连接拓扑示意图



链形连接

1.1.5 角色选举

角色选举会在以下情况下进行：

- IRF 建立。
- 主设备离开或者故障。
- IRF 分裂。
- 独立运行的两个（或多个）IRF 合并为一个 IRF。

说明

IRF 分裂后重新合并时不进行角色选举，此时主设备的确定方式请参见 1.1.9 3. MAD 故障恢复。

角色选举中按照如下优先级顺序选择主设备：

(1) 当前的主设备优先，即 IRF 不会因为有新的成员设备加入而重新选举主设备即使新的成员设备有更高优先级。该规则不适用于 IRF 形成时，此时所有加入的设备都认为自己是主设备。

(2) 成员优先级大的设备。

(3) 系统运行时间长的设备。在 IRF 中，运行时间的度量精度为 10 分钟，即如果设备的启动时间间隔小于等于 10 分钟，则认为它们运行时间相等。

(4) CPU MAC 地址小的设备。

通过以上规则选出的最优成员设备即为主设备，其它成员设备均为从设备。

IRF 建立时，所有从设备必须重启加入 IRF。

独立运行的 IRF 合并时，竞选失败方的所有成员设备必须重启加入获胜方。

1.1.6 IRF 中的接口命名规则

对于独立运行的设备（即没有加入任何 IRF），接口编号采用槽位编号/子槽位编号/接口序号的三维格式。

例如，要将独立运行的设备 Sysname 的接口 GigabitEthernet1/0/1 的链路类型设置为 Trunk，可参照以下步骤：

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

```
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] port link-type trunk
```

对于 IRF 中的成员设备，接口编号采用成员设备编号/槽位编号/子槽位编号/接口序号的四维格式。例如，将成员编号为 1 的设备上 2 槽位第一个端口的链路类型设置为 Trunk，可参照以下步骤：

```
<Sysname> system-view
[Sysname] interface gigabitethernet 1/2/0/1
[Sysname-GigabitEthernet1/2/0/1] port link-type trunk
```

1.1.7 IRF 中的文件系统命名规则

对于独立运行的设备，直接使用存储介质的名称可以访问主用主控板的文件系统，使用“slotMember-ID#存储介质的名称”可以访问备用主控板的文件系统。存储介质的命名请参见“基础配置指导”中的“文件系统管理”。

对于 IRF 中的成员设备，直接使用存储介质的名称可以访问全局主用主控板的文件系统，使用“chassisID#slotMember-ID#存储介质的名称”可以访问全局备用设备的文件系统。例如：

- 创建并显示 IRF 中全局主用主控板存储介质 Flash 根目录下的 test 文件夹：

```
<Master> mkdir test
Creating directory flash:/test... Done.
<Master> cd test
<Master> dir
Directory of flash:/test
The directory is empty.

524288 KB total (29832 KB free)
```

- 创建并显示 IRF 中成员编号为 1 的从设备上 0 槽位主控板存储介质 Flash 根目录下的 test 文件夹：

```
<Master> mkdir chassis1#slot0#flash:/test
Creating directory chassis1#slot0#flash:/test... Done.
<Master> cd chassis1#slot0#flash:/test
<Master> dir
Directory of chassis1#slot0#flash:/test
The directory is empty.

524288 KB total (128812 KB free)
```

1.1.8 IRF 中的配置同步

IRF 技术使用了严格的配置同步机制，来保证 IRF 中的多台设备能够像一台设备一样在网络中工作，并且在主设备出现故障之后，其余设备仍能够正常执行各项功能。

IRF 中的配置同步包括批量同步和实时同步两个阶段：

(1) 批量同步

- 当新设备加入 IRF 时，新设备作为从设备角色重启。新设备启动过程中，新设备会将全局主用主控板的当前配置同步到本地主控板并执行，但如下三条命令会以新设备上的配置为准，并且新设备会将这三条命令同步给主设备：
  - **irf member description**

  - **irf member priority**
  - **irf-port**
  - 新设备上的原配置文件仍然存在，但不再生效，除非该设备恢复到独立运行模式。
- 当整个 IRF 系统重启时，IRF 中的所有设备同时启动，则从设备会将全局主用主控板的启动配置文件同步至本地主控板并执行。

- 实时同步：在 IRF 正常工作后，用户所进行的任何配置，都会记录到全局主用主控板的当前配置中，并同步到 IRF 中的各个全局备用主控板执行。

通过批量和实时同步，IRF 中所有主控板均运行相同的配置，即使主设备/全局主用主控板出现故障，其它设备仍能够按照相同的配置文件执行各项功能。请根据需要执行 **save all** 命令，IRF 会将当前运行配置保存到所有主控板的存储介质上，以免 IRF 系统重启后配置丢失。

1.1.9 MAD 功能

IRF 链路故障会导致一个 IRF 变成多个新的 IRF。这些 IRF 拥有相同的 IP 地址等三层配置，会引起地址冲突，导致故障在网络中扩大。为了提高系统的可用性，当 IRF 分裂时我们就需要一种机制，能够检测出网络中同时存在多个 IRF，并进行相应的处理，尽量降低 IRF 分裂对业务的影响。MAD（Multi-Active Detection，多 Active 检测）就是这样一种检测和处理机制。MAD 主要提供分裂检测、冲突处理和故障恢复功能。

1. 分裂检测

通过 LACP（Link Aggregation Control Protocol，链路聚合控制协议）、BFD（Bidirectional Forwarding Detection，双向转发检测）、ARP（Address Resolution Protocol，地址解析协议）或者 ND（Neighbor Discovery，邻居发现）来检测网络中是否存在多个 IRF。同一 IRF 中可以配置一个或多个检测机制，详细信息，请参考“1.1.10 MAD 检测机制”。

关于 LACP 的详细介绍请参见“二层技术-以太网交换配置指导”中的“以太网链路聚合”；关于 BFD 的详细介绍请参见“网络管理和监控配置指导”中的“BFD”；关于 ARP 的详细介绍请参见“三层技术-IP 业务配置指导”中的“ARP”；关于 ND 的详细介绍请参见“三层技术-IP 业务配置指导”中的“IPv6 基础”。

2. 冲突处理

IRF 分裂后，通过分裂检测机制 IRF 会检测到网络中存在其它处于正常工作状态的 IRF。

- 对于 LACP MAD 和 BFD MAD 检测，冲突处理会先比较两个 IRF 中成员设备的数量，数量多的 IRF 继续工作，数量少的迁移到 Recovery 状态（即禁用状态）。如果成员数量相等，则主设备成员编号小的 IRF 继续工作，其它 IRF 迁移到 Recovery 状态。
- 对于 ARP MAD 和 ND MAD 检测，冲突处理会直接让主设备成员编号小的 IRF 继续工作；其它 IRF 迁移到 Recovery 状态。

IRF 迁移到 Recovery 状态后会关闭该 IRF 中所有成员设备上除保留端口以外的其它所有业务端口，以保证该 IRF 不能再转发业务报文。保留端口可通过 **mad exclude interface** 命令配置。

3. MAD 故障恢复

IRF 链路故障导致 IRF 分裂，从而引起多 Active 冲突。因此修复故障的 IRF 链路，让冲突的 IRF 重新合并为一个 IRF，就能恢复 MAD 故障。

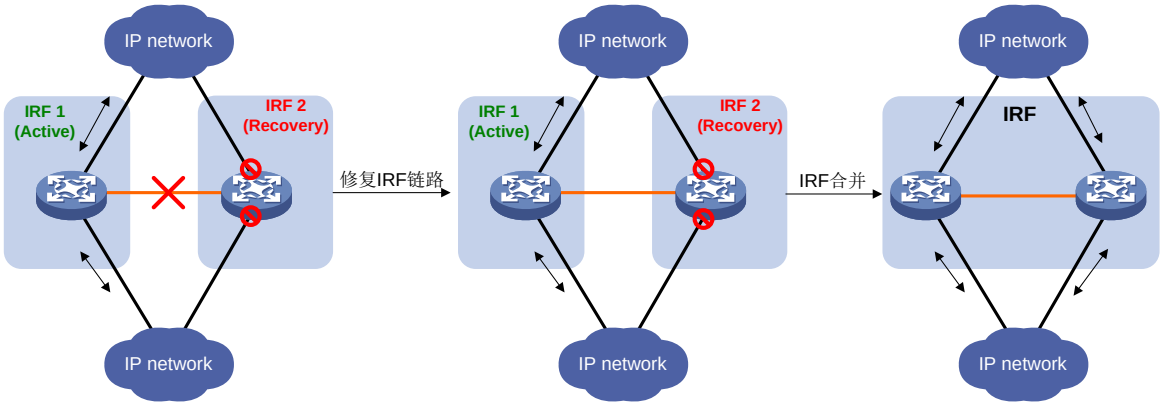
IRF 链路修复后，系统会自动重启或者给出提示信息要求用户手工重启处于 Recovery 状态的 IRF。

重启后，原 Recovery 状态 IRF 中所有成员设备以从设备身份加入原正常工作状态的 IRF，原 Recovery 状态 IRF 中被强制关闭的业务接口会自动恢复到真实的物理状态，整个 IRF 系统恢复，如图 1-7 所示。

注意

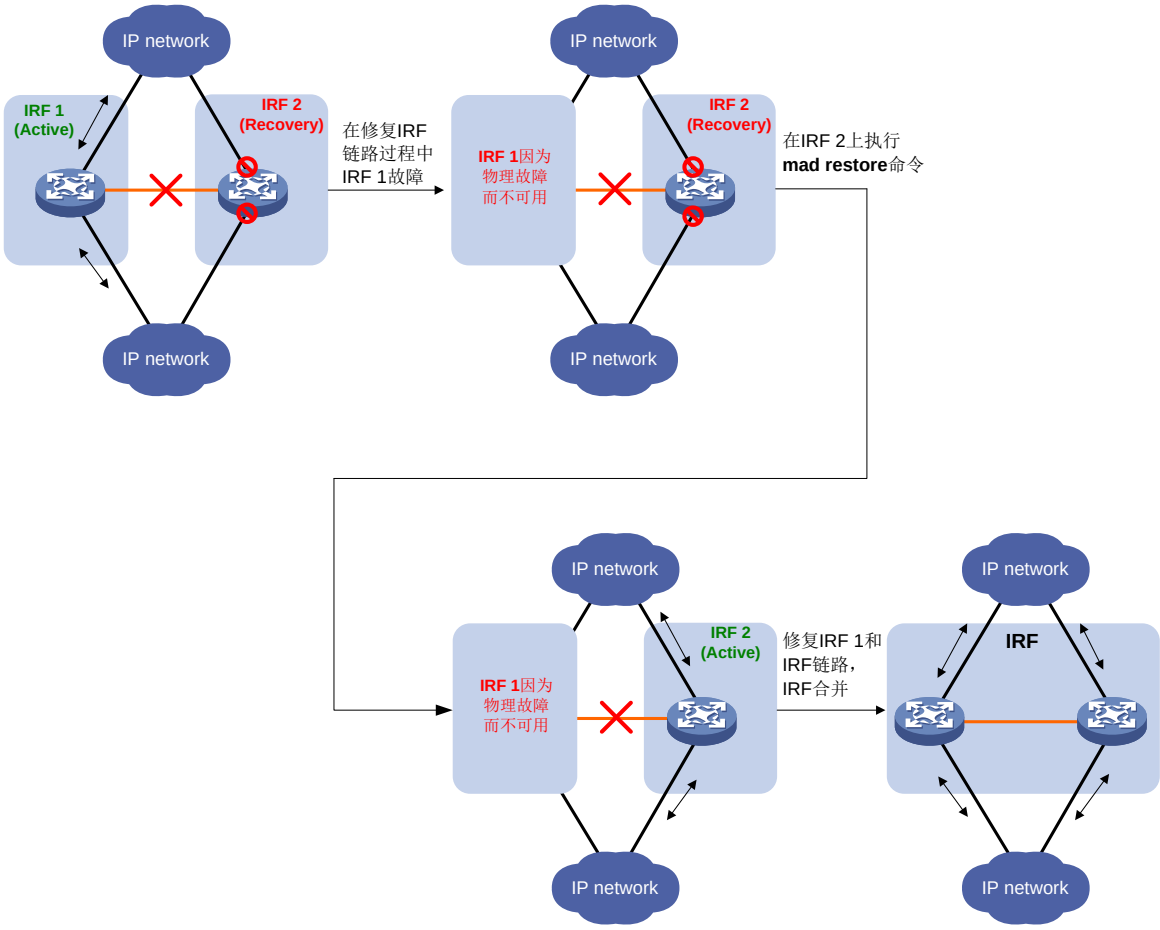
- 系统是否会自动重启或者给出提示信息要求用户手工重启处于 Recovery 状态的 IRF，与设备是否配置了 **irf auto-merge enable** 命令有关。
- 请根据提示重启处于 Recovery 状态的 IRF，如果错误的重启了正常工作状态的 IRF，会导致合并后的 IRF 仍然处于 Recovery 状态，所有成员设备的业务接口都会被关闭。此时，需要执行 **mad restore** 命令让整个 IRF 系统恢复。

图1-7 MAD 故障恢复（IRF 链路故障）



如果 MAD 故障还没来得及恢复而处于正常工作状态的 IRF 也故障了（原因可能是设备故障或者上下行线路故障），如图 1-8 所示。此时可以在 Recovery 状态的 IRF 上执行 **mad restore** 命令，让 Recovery 状态的 IRF 恢复到正常状态，先接替原正常工作状态的 IRF 工作。然后再修复故障的 IRF 和链路。

图1-8 MAD 故障恢复（IRF 链路故障修复前，正常工作状态的 IRF 故障）



1.1.10 MAD 检测机制

设备支持的 MAD 检测方式有：LACP MAD 检测、BFD MAD 检测、ARP MAD 检测和 ND MAD 检测。四种 MAD 检测机制各有特点，用户可以根据现有组网情况进行选择。

表1-2 MAD 检测机制的比较

MAD 检测方式	优势	限制	适用组网
LACP MAD	• 检测速度快 • 利用现有聚合组网即可实现，无需占用额外接口	需要使用H3C设备（支持扩展LACP协议报文）作为中间设备	IRF使用聚合链路和上行设备或下行设备连接
BFD MAD	• 检测速度较快 • 使用中间设备时，不要求中间设备必须为 H3C 设备	需要专用的物理链路和三层接口，这些接口不能再传输普通业务流量	• 对组网没有特殊要求 • 如果不使用中间设备，则仅适用于成员设备少（建议仅 2 台成员设备时使用），并且物理距离比较近的组网环境

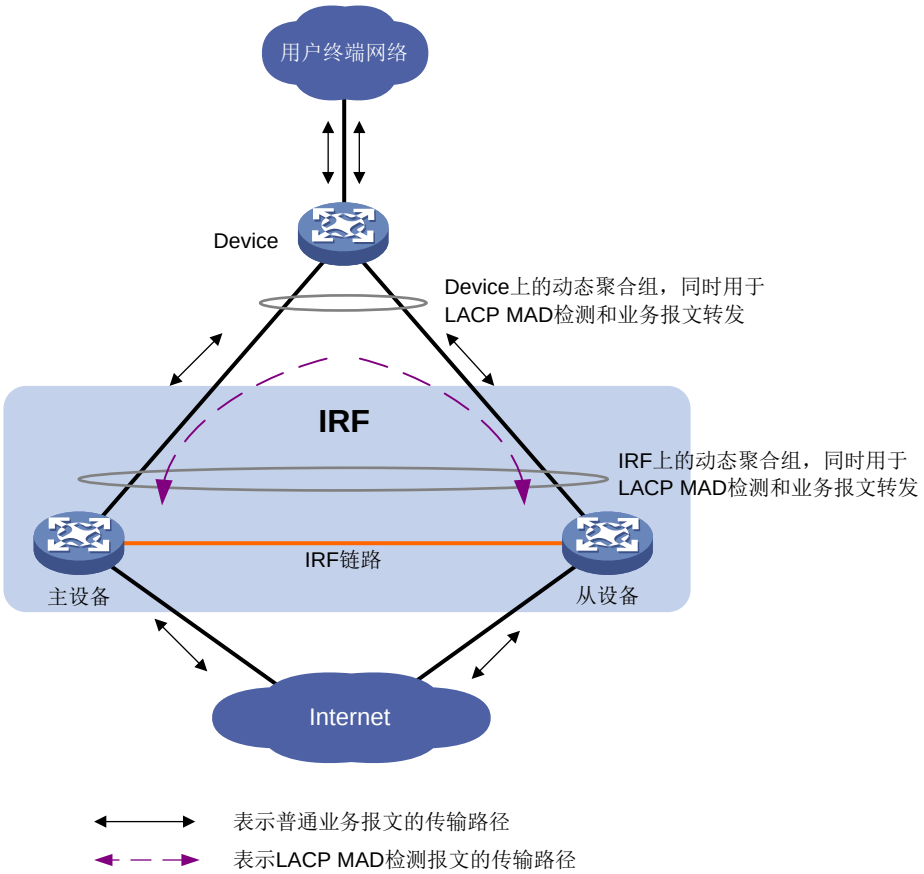
MAD 检测方式	优势	限制	适用组网
ARP MAD	• 可以不使用中间设备 • 使用中间设备时，不要求中间设备必须为 H3C 设备 • 无需占用额外接口	• 检测速度慢于 LACP MAD 和 BFD MAD • 必须和生成树协议配合使用	适用于使用生成树，没有使用链路聚合的IPv4组网环境
ND MAD	• 可以不使用中间设备 • 使用中间设备时，不要求中间设备必须为 H3C 设备 • 无需占用额外接口	• 检测速度慢于 LACP MAD 和 BFD MAD • 必须和生成树协议配合使用	适用于使用生成树，没有使用链路聚合的IPv6组网环境

1. LACP MAD 检测

LACP MAD 检测通过扩展 LACP 协议报文实现，通常采用如图 1-9 所示的组网：

- 每个成员设备都需要连接到中间设备。
- 成员设备连接中间设备的链路加入动态聚合组。
- 中间设备需要支持扩展 LACP 报文。

图1-9 LACP MAD 检测组网示意图



扩展 LACP 协议报文定义了一个新的 TLV（Type/Length/Value，类型/长度/值）数据域——用于交互 IRF 的 DomainID（域编号）和 ActiveID（主设备的成员编号）。开启 LACP MAD 检测后，成员设备通过 LACP 协议报文和其它成员设备交互 DomainID 和 ActiveID 信息。

- 如果 DomainID 不同，表示报文来自不同 IRF，不需要进行 MAD 处理。
- 如果 DomainID 相同，ActiveID 也相同，表示没有发生多 Active 冲突。
- 如果 DomainID 相同，ActiveID 不同，表示 IRF 分裂，检测到多 Active 冲突。

2. BFD MAD 检测

BFD MAD 检测通过 BFD 协议实现。我们可以使用以太网端口来实现 BFD MAD 检测。

使用以太网端口实现 BFD MAD 时，请注意如下组网要求：

- 使用中间设备时（如图 1-10 所示），每台成员设备都需要和中间设备建立 BFD MAD 检测链路。不使用中间设备时，每台成员设备必须和其它所有成员设备之间建立 BFD MAD 检测链路（如图 1-11 所示）。
- 用于 BFD MAD 检测的以太网端口加入同一三层聚合组，在该三层聚合接口视图下为每台成员设备配置 MAD IP 地址。

需要注意的是：

- BFD MAD 检测链路和 BFD MAD 检测 VLAN 或 BFD MAD 检测三层聚合接口必须是专用的，不允许配置任何其它特性。
- MAD IP 地址应该为同一网段内的不同 IP 地址。
- 两台以上设备组成 IRF 时，请优先采用中间设备组网方式，避免特殊情况下全连接组网中可能出现的广播环路问题。
- 使用三层聚合接口配置 BFD MAD 时，聚合成员端口的个数不能超过聚合组最大选中端口数。否则，由于超出聚合组最大选中端口数的成员端口无法成为选中端口，会使 BFD MAD 无法正常工作，工作状态显示为 Faulty。有关聚合组最大选中端口的说明及其配置方式请参见“二层技术-以太网交换配置指导”中的“以太网链路聚合”。

图1-10 使用中间设备实现 BFD MAD 检测组网示意图

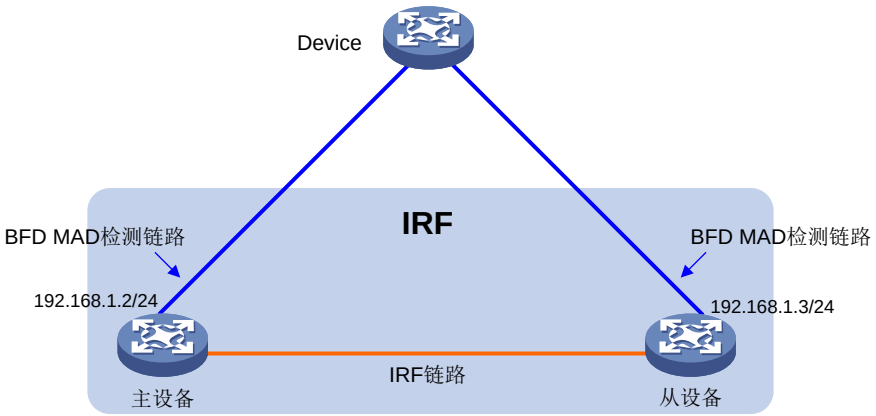
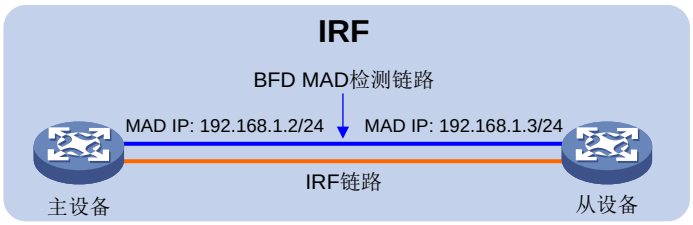


图1-11 不使用中间设备实现 BFD MAD 检测组网示意图



BFD MAD 实现原理如下：

- 当 IRF 正常运行时，只有主设备上配置的 MAD IP 地址生效，从设备上配置的 MAD IP 地址不生效，BFD 会话处于 down 状态；（使用 **display bfd session** 命令查看 BFD 会话的状态。如果 Session State 显示为 Up，则表示激活状态；如果显示为 Down，则表示处于 down 状态）。
- 当 IRF 分裂形成多个 IRF 时，不同 IRF 中主设备上配置的 MAD IP 地址均会生效，BFD 会话被激活，此时会检测到多 Active 冲突。

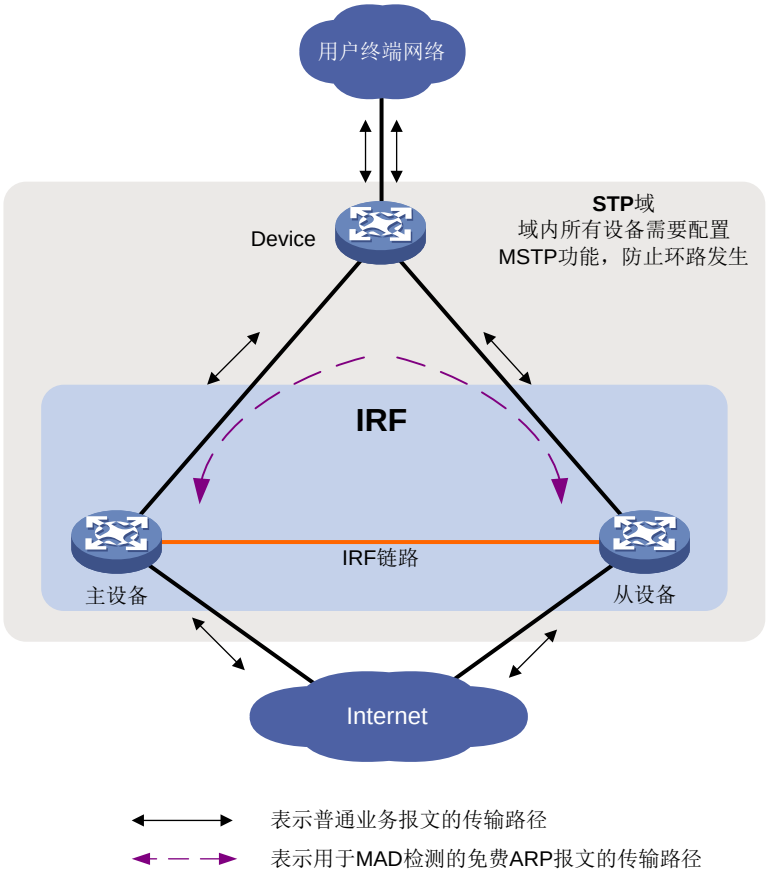
3. ARP MAD 检测

ARP MAD 检测是通过使用扩展 ARP 协议报文交互 IRF 的 DomainID 和 ActiveID 实现的。

配置 ARP MAD 时，可以使用中间设备，也可以不使用中间设备。

- 使用中间设备时，每台成员设备都需要和中间设备建立连接，如图 1-12 所示。IRF 和中间设备之间需要运行生成树协议。可以使用数据链路作为 ARP MAD 检测链路。
- 不使用中间设备时，每台成员设备必须和其它所有成员设备之间建立 ARP MAD 检测链路。

图1-12 ARP MAD 检测组网示意图



开启 ARP MAD 检测后，成员设备通过 ARP 协议报文和其它成员设备交互 DomainID 和 ActiveID 信息。

- 如果 DomainID 不同，表示报文来自不同 IRF，不需要进行 MAD 处理。
- 如果 DomainID 相同，ActiveID 也相同，表示没有发生多 Active 冲突。
- 如果 DomainID 相同，ActiveID 不同，表示 IRF 分裂，检测到多 Active 冲突。

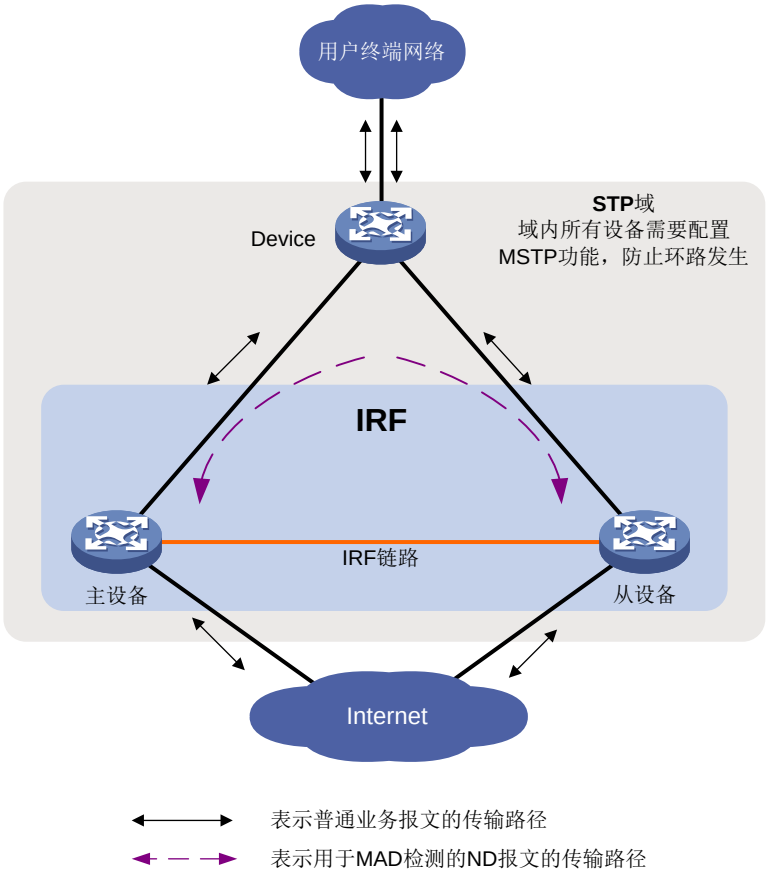
4. ND MAD 检测

ND MAD 检测是通过扩展 ND 协议报文内容实现的，即使用 ND 的 NS 协议报文携带扩展选项数据来交互 IRF 的 DomainID 和 ActiveID。

配置 ND MAD 时，可以使用中间设备，也可以不使用中间设备。

- 使用中间设备时，每台成员设备都需要和中间设备建立连接，如图 1-13 所示。IRF 和中间设备之间需要运行生成树协议。
- 不使用中间设备时，每台成员设备必须和其它所有成员设备之间建立 ND MAD 检测链路。

图1-13 ND MAD 检测组网示意图



开启 ND MAD 检测后，成员设备通过 ND 协议报文和其它成员设备交互 DomainID 和 ActiveID 信息。

- 如果 DomainID 不同，表示报文来自不同 IRF，不需要进行 MAD 处理。
- 如果 DomainID 相同，ActiveID 也相同，表示没有发生多 Active 冲突。
- 如果 DomainID 相同，ActiveID 不同，表示 IRF 分裂，检测到多 Active 冲突。

1.2 IRF与硬件适配关系

本特性的支持情况与设备型号有关，请以设备的实际情况为准。

设备型号	业务板类型	说明
M9006 M9010 M9014	Blade IV防火墙业务板	支持
	Blade V防火墙业务板	支持
	NAT业务板	支持
M9010-GM	加密业务板	支持
M9016-V	Blade V防火墙业务板	支持
M9008-S	Blade IV防火墙业务板	支持

设备型号	业务板类型	说明
M9012-S	入侵防御业务板	支持
	视频网关业务板	支持
M9008-S-V	Blade IV防火墙业务板	支持
M9000-AI-E4 M9000-AI-E8 M9000-AI-E16 M9000-AK001	Blade V防火墙业务板	支持
M9000-X06 M9000-X06-B M9000-X06-B-G M9000-X06-G M9000-X10	Blade VI防火墙业务板	不支持
M9000-AI-X06 M9000-AI-X10	Blade VI防火墙业务板	不支持

1.3 IRF配置限制和指导

1.3.1 硬件兼容性相关配置限制和指导

通常情况下，必须是同一型号的产品才能组成 IRF，且相同槽位号上插入的板卡型号也必须相同。

1.3.2 软件版本要求

IRF 中所有成员设备的软件版本必须相同，如果有软件版本不同的设备要加入 IRF，请确保 IRF 的启动文件同步加载功能处于开启状态。

1.3.3 IRF 规模

一个 IRF 中允许加入的成员设备的最大数量为 2。

1.3.4 确定 IRF 物理端口

通常情况下，要求是设备上的高速率端口作为 IRF 物理端口。

设备出厂时没有将 IRF 端口与 IRF 物理端口绑定，需要用户通过命令行手工配置后才能用于 IRF。

NSQM1CGQ20 的 100G 接口与 NSQ1CGC2SE0 的 100G 接口无法使用 IRF。

不能将 Console 口、管理口（面板标识为 MGMT，编号为 M-GigabitEthernet 开头）和配置了 Bypass 功能或面板标识了 Bypass 功能的接口作为 IRF 物理端口。

1.3.5 选择连接 IRF 端口的模块

设备支持可插拔接口模块，您可根据单板接口支持的可插拔接口模块类型，选择相应的可插拔接口模块，具体如下：

- 万兆 XFP 模块。
- 万兆 SFP+模块。
- QSFP+模块。
- 100G CFP 模块。
- QSFP28 模块。

说明

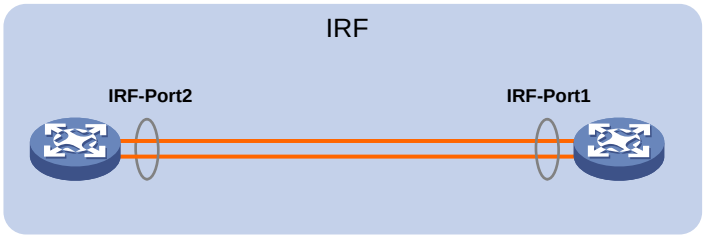
- 有关光模块和电缆的详细介绍，请参见《H3C 光模块手册》。
- H3C 光模块和电缆的种类随着时间变化有更新的可能性，所以，若您需要准确的模块种类信息，请咨询 H3C 公司市场人员或技术支援人员。

1.3.6 IRF 物理端口连接要求

本设备上与 IRF-Port1 口绑定的 IRF 物理端口只能和邻居成员设备 IRF-Port2 口上绑定的 IRF 物理端口相连，本设备上与 IRF-Port2 口绑定的 IRF 物理端口只能和邻居成员设备 IRF-Port1 口上绑定的 IRF 物理端口相连，如图 1-14 所示。否则，不能形成 IRF。

一个 IRF 端口可以与一个或多个 IRF 物理端口绑定，以提高 IRF 链路的带宽以及可靠性。设备仅支持 IRF 物理端口直连组建 IRF，不支持跨中间设备。

图1-14 IRF 物理连接示意图



1.3.7 IRF 物理端口配置限制和指导

1. IRF 物理端口配置限制

以太网接口作为 IRF 物理端口与 IRF 端口绑定后，只支持配置以下命令：

- 接口配置命令，包括 **shutdown**、**description** 和 **flow-interval** 命令。有关这些命令的详细介绍，请参见“接口管理命令参考”中的“以太网接口”。
- 配置光模块的 ITU 通道编号 **itu-channel**。有关 **itu-channel** 命令的详细介绍，请参见“基础配置命令参考”中的“设备管理”。

- LLDP 功能命令，包括 **lldp admin-status**、**lldp check-change-interval**、**lldp enable**、**lldp encapsulation snap**、**lldp notification remote-change enable** 和 **lldp tlv-enable**。有关这些命令的详细介绍，请参见“二层技术-以太网交换命令参考”中的“LLDP”。
- 将端口配置为远程源镜像反射端口，**mirroring-group reflector-port** 命令，但配置后端口与 IRF 端口绑定的配置将被清除。当 IRF 端口只绑定了一个物理端口时请勿进行此配置，以免 IRF 分裂。有关该命令的详细介绍，请参见“网络管理和监控命令参考”中的“端口镜像”。

2. IRF 物理端口的环路避免与 SNMP 监测

IRF 成员设备根据接收和发送报文的端口以及 IRF 的当前拓扑，来判断报文发送后是否会产生环路。如果判断结果为会产生环路，设备将在环路路径的发送端口处将报文丢弃。该方式会造成大量广播报文在 IRF 物理端口上被丢弃，此为正常现象。在使用 SNMP 工具监测设备端口的收发报文记录时，取消对 IRF 物理端口的监测，可以避免收到大量丢弃报文的告警信息。

1.3.8 IRF 与其它软件特性的兼容性与限制

1. 路由

在组成 IRF 的所有设备上，以下路由相关配置必须相同，否则这些设备将无法组成 IRF。

- 等价路由模式（通过 **ecmp mode** 命令配置）。
- 前缀大于 64 位的 IPv6 路由功能。

关于上述功能的详细介绍，请参见“三层技术-IP 路由配置指导”中的“IP 路由基础配置”。

1.3.9 IRF 中 License 安装一致性要求

请确保 IRF 中各成员设备上安装的特性 License 一致，否则，可能会导致这些 License 对应的特性不能正常运行。

1.3.10 配置回滚限制

以下 IRF 相关配置不支持配置回滚：

- 配置成员设备的描述信息（**irf member description**）
- 配置 IRF 中成员设备的优先级（**irf member priority**）
- 配置 IRF 端口与 IRF 物理端口的绑定关系（**port group interface**）

有关配置回滚的详细介绍，请参见“基础配置指导”中的“配置文件”。

1.4 IRF配置任务简介

IRF 配置任务如下：

(1) 搭建 IRF

a. 配置成员编号

b. （可选）配置成员优先级

c. 配置 IRF 端口

d. 将当前配置保存到设备的下次启动配置文件

e. 连接 IRF 物理接口

f. 切换到 IRF 模式

g. 访问 IRF

(2) 配置 MAD

请至少选择其中一项 MAD 检测方案进行配置。

- 配置 LACP MAD 检测
- 配置 BFD MAD 检测
- 配置 ARP MAD 检测
- 配置 ND MAD 检测
- 配置保留接口

  IRF 迁移到 Recovery 状态后会关闭该 IRF 中除保留接口以外的所有业务接口。如果接口有特殊用途需要保持 up 状态（比如 Telnet 登录接口），可以将这些接口配置为保留接口。
- MAD 故障恢复

(3) （可选）调整和优化 IRF

- 配置成员编号
- 配置成员优先级
- 配置 IRF 端口
- 快速配置 IRF 基本参数

  可以分别调整成员编号、成员优先级、IRF 端口，也可以使用本功能同时调整这三个参数。
- 开启 IRF 合并自动重启功能

  IRF 合并时，竞选失败方 IRF 的所有成员设备自动重启加入获胜方 IRF。
- 配置成员设备的描述信息
- 配置 IRF 链路的负载分担模式
- 配置 IRF 的桥 MAC 地址
- 开启启动文件的自动加载功能

  新设备/新主控板加入 IRF，且新设备/新主控板的软件版本和全局主用主控板的软件版本不一致时，新设备的主控板/新主控板自动从全局主用主控板下载启动文件，然后使用新的系统启动文件重启，重新加入 IRF。
- 配置 IRF 链路 down 延迟上报功能

1.5 配置准备

进行网络规划，确定以下项目：

- 硬件兼容性和限制（选择哪些型号的设备，是否要求同型号）
- IRF 规模（包含几台成员设备）
- 使用哪台设备作为主设备
- 各成员设备编号和优先级分配方案。IRF 形成后，尽量不要修改成员编号。
- IRF 拓扑和物理连接方案

- 确定 IRF 物理端口

1.6 搭建IRF

1.6.1 配置成员编号

1. 功能简介

出厂时，设备处于独立运行模式，没有成员编号。设备从独立运行模式切换到 IRF 模式时，使用本功能配置的成员编号。如果模式切换前未配置成员编号，则系统自动使用 1 作为成员编号。

建议在切换为 IRF 模式前先配置成员编号，并确保该编号在 IRF 中唯一。如果存在成员编号相同的设备，则不能建立 IRF。如果新设备加入 IRF，但是该设备与已有成员设备的编号冲突，则该设备不能加入 IRF。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 在独立运行模式下配置设备的成员编号。

irf member *member-id*

缺省情况下，设备处于独立运行模式，没有成员编号。

1.6.2 配置成员优先级

1. 功能简介

在主设备选举过程中，优先级数值大的成员设备将优先被选举成为主设备。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 在独立运行模式下配置设备的成员优先级。

irf priority *priority*

缺省情况下，设备的成员优先级为 1。

1.6.3 配置 IRF 端口

1. 功能简介

在独立运行模式下将 IRF 端口和 IRF 物理端口绑定，并不会影响 IRF 物理端口的当前业务。当设备切换到 IRF 模式后，IRF 物理端口的配置将恢复到缺省状态（即原有的业务配置会被删除）。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 在独立运行模式下创建 IRF 端口并进入 IRF 端口视图。

irf-port *irf-port-number*

(3) 将 IRF 端口和 IRF 物理端口绑定。

port group interface *interface-type interface-number*

缺省情况下，IRF 端口没有和任何 IRF 物理端口绑定。

多次执行本命令，可以将 IRF 端口与多个 IRF 物理端口绑定，以实现 IRF 链路的备份/负载分担。

1.6.4 将当前配置保存到设备的下次启动配置文件

请在任意视图下执行本命令，将当前配置保存到存储介质的根目录下，并将该文件设置为下次启动配置文件。

save

有关该命令的详细介绍，请参见“基础配置命令参考”中的“配置文件管理”。

1.6.5 连接 IRF 物理接口

请按照拓扑规划和“1.3.6 IRF 物理端口连接要求”完成 IRF 物理端口连接。

1.6.6 切换到 IRF 模式

1. 功能简介

设备缺省处于独立运行模式。要使设备加入 IRF 或使设备的 IRF 配置生效，必须将设备运行模式切换到 IRF 模式。

修改运行模式后，设备会自动重启使新的模式生效。

2. 配置限制和指导

模式切换会导致配置不可用。为了使当前配置在模式切换后能够尽可能多的继续生效，在用户执行模式切换操作时，系统会提示用户是否需要自动转换下次启动配置文件。如果用户选择了<Y>，则设备会自动对下次启动配置文件进行转换和保存。对于 E1、E3、T1、T3 接口，配置转换的匹配规则为 *interface-type* 加空格加 *interface-number*；对于其他类型接口，配置转换的匹配规则为 *interface-type* 加 *interface-number*（两参数中间不允许空格）。配置转换后，*interface-number* 会增加一维，第一维为成员编号，其它维度的取值和转换前的取值一致。需要注意的是，如果用户配置的字符串类型参数（**description** 命令配置的字符串类型参数除外）符合上述转换规则，该参数也会被转换。例如当有 VLAN 的名称被定义为 GigabitEthernet1/0/7 时，该名称也会被转换。

3. 配置准备

在切换到 IRF 模式前，请先配置成员编号，并确保该编号在 IRF 中唯一。

4. 配置步骤

(1) 进入系统视图。

system-view

(2) 将设备的运行模式切换到 IRF 模式。

chassis convert mode irf

缺省情况下，设备处于独立运行模式。

因为管理和维护 IRF 需要耗费一定的系统资源。如果当前组网中设备不需要和别的设备组成 IRF 时，请执行 **undo chassis convert mode**，将 IRF 模式切换到独立运行模式。

1.6.7 访问 IRF

完成 IRF 模式切换，设备重启后，可通过如下方式登录 IRF：

- 本地登录：通过任意成员设备的 Console 口登录。
- 远程登录：给任意成员设备的任意三层接口配置 IP 地址，并且路由可达，就可以通过 Telnet、SNMP 等方式进行远程登录。

不管使用哪种方式登录 IRF，实际上登录的都是全局主用主控板。全局主用主控板是 IRF 系统的配置和控制中心，在全局主用主控板上配置后，全局主用主控板会将相关配置同步给全局备用主控板，以便保证全局主用主控板和全局备用主控板配置的一致性。

1.7 配置MAD

1.7.1 配置限制和指导

1. IRF 域编号配置指导

IRF 域编号是一个全局变量，IRF 中的所有成员设备都共用这个 IRF 域编号。在 IRF 设备上使用 **irf domain**、**mad enable**、**mad arp enable**、**mad nd enable** 命令均可修改全局 IRF 域编号，最新的配置生效。请按照网络规划来修改 IRF 域编号，不要随意修改。

在 LACP MAD、ARP MAD 和 ND MAD 检测组网中，如果中间设备本身也是一个 IRF 系统，则必须通过配置确保其 IRF 域编号与被检测的 IRF 系统不同，否则可能造成检测异常，甚至导致业务中断。在 BFD MAD 检测组网中，IRF 域编号为可选配置。

2. 被 MAD 关闭的接口恢复指导

如果接口因为多 Active 冲突被关闭，则只能等 IRF 恢复到正常工作状态后，接口才能自动被激活，不允许通过 **undo shutdown** 命令来激活，否则可能引起配置冲突，导致故障在网络中扩大。

1.7.2 配置 LACP MAD 检测

(1) 进入系统视图。

system-view

(2) 配置 IRF 域编号。

irf domain *domain-id*

缺省情况下，IRF 的域编号为 0。

注意

修改设备的 IRF 域编号，会导致设备离开当前 IRF，不再属于当前 IRF，不能和当前 IRF 中的设备交互 IRF 协议报文。

---

(3) 创建并进入聚合接口视图。请选择其中一项进行配置。

- 进入二层聚合接口视图。

interface bridge-aggregation *interface-number*

- 进入三层聚合接口视图。

  interface route-aggregation *interface-number*

中间设备上也需要进行此项配置。

(4) 配置聚合组工作在动态聚合模式下。

link-aggregation mode dynamic

缺省情况下，聚合组工作在静态聚合模式下。

中间设备上也需要进行此项配置。

(5) 开启 LACP MAD 检测功能。

mad enable

缺省情况下，LACP MAD 检测功能处于关闭状态。

(6) 退回系统视图。

quit

(7) 进入以太网接口视图。

interface *interface-type interface-number*

(8) 将以太网接口加入聚合组。

port link-aggregation group *group-id*

中间设备上也需要进行此项配置。

1.7.3 配置 BFD MAD 检测

1. 配置限制和指导

使用三层聚合接口进行 BFD MAD 检测时，请注意表 1-3 所列配置注意事项。

表1-3 使用三层聚合接口进行 BFD MAD 检测

注意事项类别	使用限制和注意事项
三层聚合接口配置	• 必须使用静态聚合模式的三层聚合接口（聚合接口缺省工作在静态聚合模式） • 聚合成员端口的个数不能超过聚合组最大选中端口数。否则，由于超出聚合组最大选中端口数的成员端口无法成为选中端口，会使 BFD MAD 无法正常工作，工作状态显示为 Faulty
BFD MAD检测VLAN	• 如果使用中间设备，请将中间设备上用于 BFD MAD 检测的物理接口添加到同一个 VLAN 中，并允许 PVID 的报文不带 Tag 通过。中间设备上的端口不用加入聚合组 • 如果设备充当多个 IRF BFD MAD 检测的中间设备，请为各 IRF 分配不同的 VLAN • 中间设备上用于 BFD MAD 检测的 VLAN 必须专用，不允许运行其他业务。且该 VLAN 中只能包含 BFD MAD 检测链路上的端口，请不要将其它端口加入该 VLAN。当某个业务端口需要使用 **port trunk permit vlan all** 命令允许所有 VLAN 通过时，请使用 **undo port trunk permit** 命令将用于 BFD MAD 的 VLAN 排除
开启BFD MAD检测功能的三层聚合接口的特性限制	开启BFD MAD检测功能的接口只能配置**mad bfd enable**和**mad ip address**命令。如果用户配置了其它业务，可能会影响该业务以及BFD MAD检测功能的运行

注意事项类别	使用限制和注意事项
MAD IP地址	• 在用于 BFD MAD 检测的接口下必须使用 **mad ip address** 命令配置 MAD IP 地址，而不要配置其它 IP 地址（包括使用 **ip address** 命令配置的普通 IP 地址、VRRP 虚拟 IP 地址等），以免影响 MAD 检测功能 • 为不同成员设备配置同一网段内的不同 MAD IP 地址

2. 使用三层聚合接口进行 BFD MAD 检测配置步骤

(1) 进入系统视图。

system-view

(2) （可选）配置 IRF 域编号。

irf domain *domain-id*

缺省情况下，IRF 的域编号为 0。

注意

修改设备的 IRF 域编号，会导致设备离开当前 IRF，不再属于当前 IRF，不能和当前 IRF 中的设备交互 IRF 协议报文。

(3) 创建一个新三层聚合接口专用于 BFD MAD 检测。

interface route-aggregation *interface-number*

(4) 退回系统视图。

quit

(5) 进入以太网接口视图。

interface *interface-type interface-number*

(6) 将端口加入 BFD MAD 检测专用聚合组。

port link-aggregation group *number*

(7) 退回系统视图。

quit

(8) 进入三层聚合接口视图。

interface route-aggregation *interface-number*

(9) 开启 BFD MAD 检测功能。

mad bfd enable

缺省情况下，BFD MAD 检测功能处于关闭状态。

(10) 给指定成员设备配置 MAD IP 地址。

mad ip address *ip-address* { *mask* | *mask-length* } **member** *member-id*

缺省情况下，未配置成员设备的 MAD IP 地址。

1.7.4 配置 ARP MAD 检测

1. 配置限制和指导

配置 ARP MAD 检测时，请注意表 1-4 所列配置注意事项。

表1-4 ARP MAD 检测配置注意事项

注意事项类别	使用限制和注意事项

ARP MAD检测VLAN	• 不允许在 Vlan-interface1 接口上开启 ARP MAD 检测功能
	• 如果使用中间设备，需要进行如下配置：
	  ○ 在 IRF 设备和中间设备上，创建专用于 ARP MAD 检测的 VLAN
	  ○ 在 IRF 设备和中间设备上，将用于 ARP MAD 检测的物理接口添加到 ARP MAD 检测专用 VLAN 中
	  ○ 在 IRF 设备上，创建 ARP MAD 检测的 VLAN 的 VLAN 接口
	• 建议勿在 ARP MAD 检测 VLAN 上运行其它业务

兼容性配置指导	如果使用中间设备，请确保满足如下要求： • IRF 和中间设备上均需配置生成树功能。并确保配置生成树功能后，只有一条 ARP MAD 检测链路处于转发状态。关于生成树功能的详细介绍请参见“二层技术-以太网交换配置指导”中的“生成树” • 如果中间设备本身也是一个 IRF 系统，则必须通过配置确保其 IRF 域编号与被检测的 IRF 系统不同

2. ARP MAD 检测配置步骤

(1) 进入系统视图。

system-view

(2) 配置 IRF 域编号。

irf domain *domain-id*

缺省情况下，IRF 的域编号为 0。

注意

修改设备的 IRF 域编号，会导致设备离开当前 IRF，不再属于当前 IRF，不能和当前 IRF 中的设备交互 IRF 协议报文。

(3) 将 IRF 的桥 MAC 保留时间配置为立即改变。

undo irf mac-address persistent

缺省情况下，IRF 桥 MAC 永久保留。

(4) 创建一个新 VLAN 专用于 ARP MAD 检测。

vlan *vlan-id*

缺省情况下，设备上只存在 VLAN 1。

VLAN 1 不能用于 ARP MAD 检测。

如果使用中间设备，中间设备上也需要进行此项配置。

(5) 退回系统视图。

quit

(6) 进入以太网接口视图。

interface *interface-type interface-number*

(7) 将端口加入 ARP MAD 检测专用 VLAN。

- 将 Access 端口加入 ARP MAD 检测专用 VLAN。

  port access vlan *vlan-id*

- 将 Trunk 端口加入 ARP MAD 检测专用 VLAN。

  port trunk permit vlan *vlan-id*

- 将 Hybrid 端口加入 ARP MAD 检测专用 VLAN。

  port hybrid vlan *vlan-id* { **tagged** | **untagged** }

ARP MAD 检测对检测端口的链路类型没有要求，不需要刻意修改端口的当前链路类型。缺省情况下，端口的链路类型为 Access。

如果使用中间设备，中间设备上也需要进行此项配置。

(8) 退回系统视图。

quit

(9) 进入 VLAN 接口视图。

interface vlan-interface *interface-number*

(10) 配置 IP 地址。

ip address *ip-address* { *mask* | *mask-length* }

缺省情况下，未配置 VLAN 接口的 IP 地址。

(11) 开启 ARP MAD 检测功能。

mad arp enable

缺省情况下，ARP MAD 检测功能处于关闭状态。

1.7.5 配置 ND MAD 检测

1. 配置限制和指导

- 当 ND MAD 检测组网使用中间设备进行连接时，可使用普通的数据链路作为 ND MAD 检测链路；当不使用中间设备时，需要在所有的成员设备之间建立两两互联的 ND MAD 检测链路。
- 如果使用中间设备组网，在 IRF 和中间设备上均需配置生成树功能。并确保配置生成树功能后，只有一条 ND MAD 检测链路处于转发状态，能够转发 ND MAD 检测报文。关于生成树功能的详细描述和配置请参见“二层技术-以太网交换配置指导”中的“生成树”。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 配置 IRF 域编号。

irf domain *domain-id*

缺省情况下，IRF 的域编号为 0。

注意

修改设备的 IRF 域编号，会导致设备离开当前 IRF，不再属于当前 IRF，不能和当前 IRF 中的设备交互 IRF 协议报文。

---

(3) 将 IRF 的桥 MAC 保留时间配置为立即改变。

undo irf mac-address persistent

缺省情况下 IRF 桥 MAC 永久保留。

(4) 创建一个新 VLAN 专用于 ND MAD 检测。

vlan *vlan-id*

缺省情况下，设备上只存在 VLAN 1。

VLAN 1 不能用于 ND MAD 检测。

如果使用中间设备，中间设备上也需要进行此项配置。

(5) 退回系统视图。

quit

(6) 进入以太网接口视图。

interface *interface-type interface-number*

(7) 端口加入 ND MAD 检测专用 VLAN。

- 将 Access 端口加入 ND MAD 检测专用 VLAN。

  port access vlan *vlan-id*

- 将 Trunk 端口加入 ND MAD 检测专用 VLAN。

  port trunk permit vlan *vlan-id*

- 将 Hybrid 端口加入 ND MAD 检测专用 VLAN。

  port hybrid vlan *vlan-id* { **tagged** | **untagged** }

ND MAD 检测对检测端口的链路类型没有要求，不需要刻意修改端口的当前链路类型。缺省情况下，端口的链路类型为 Access。

如果使用中间设备，中间设备上也需要进行此项配置。

(8) 退回系统视图。

quit

(9) 进入 VLAN 接口视图。

interface vlan-interface *interface-number*

(10) 配置 IPv6 地址。

ipv6 address { *ipv6-address/pre-length* | *ipv6 address pre-length* }

缺省情况下，未配置 VLAN 接口的 IPv6 地址。

(11) 开启 ND MAD 检测功能。

mad nd enable

缺省情况下，ND MAD 检测功能处于关闭状态。

1.7.6 配置保留接口

1. 功能简介

IRF 系统在进行多 Active 处理的时候，缺省情况下，会关闭 Recovery 状态 IRF 上除了系统保留接口外的所有业务接口。系统保留接口包括：

- IRF 物理端口
- 用户配置的保留聚合接口的成员接口

如果接口有特殊用途需要保持 up 状态（比如 Telnet 登录接口等），则用户可以通过命令行将这些接口配置为保留接口。

2. 配置限制和指导

使用 VLAN 接口进行远程登录时，需要将该 VLAN 接口及其对应的以太网端口都配置为保留接口。但如果在正常工作状态的 IRF 中该 VLAN 接口也处于 UP 状态，则在网络中会产生 IP 地址冲突。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 配置保留接口，当设备进入 Recovery 状态时，该接口不会被关闭。

mad exclude interface *interface-type interface-number*

缺省情况下，设备进入 Recovery 状态时会自动关闭本设备上除了系统保留接口以外的所有业务接口。

1.7.7 MAD 故障恢复

1. 功能简介

当 MAD 故障恢复时，处于 Recovery 状态的设备重启后重新加入 IRF，被 MAD 关闭的接口会自动恢复到正常状态。

如果在 MAD 故障恢复前，正常工作状态的 IRF 出现故障，可以通过配置本功能先启用 Recovery 状态的 IRF。配置本功能后，Recovery 状态的 IRF 中被 MAD 关闭的接口会恢复到正常状态，保证业务尽量少受影响。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 将 IRF 从 Recovery 状态恢复到正常工作状态。

mad restore

1.8 调整和优化IRF

1.8.1 配置成员编号

1. 配置限制和指导

在 IRF 中以成员编号标识设备，IRF 端口和成员优先级的配置也和成员编号紧密相关。所以，修改设备成员编号可能导致配置发生变化或者失效，请慎重使用。

配置成员编号时，请确保该编号在 IRF 中唯一。如果存在相同的成员编号，则不能建立 IRF。如果新设备加入 IRF，但是该设备与已有成员设备的编号冲突，则该设备不能加入 IRF。

- 修改成员编号后，但是没有重启本设备，则原编号继续生效，各物理资源仍然使用原编号来标识。
- 修改成员编号后，如果保存当前配置，重启本设备，则新的成员编号生效，需要用新编号来标识物理资源；配置文件中，只有 IRF 端口的编号以及 IRF 端口下的配置、成员优先级会继续生效，其它与成员编号相关的配置（比如普通物理接口的配置等）不再生效，需要重新配置。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 配置 IRF 中指定成员设备的成员编号。

irf member *member-id* **renumber** *new-member-id*

缺省情况下，设备切换到 IRF 模式后，使用的是独立运行模式下预配置的成员编号。

注意

在 IRF 中以设备编号标志设备，配置 IRF 端口和优先级也是根据设备编号来配置的，所以，修改设备成员编号可能导致设备配置发生变化或者丢失，请慎重处理。

(3) 保存当前配置。

save [**safely** | **force**]

(4) 退回用户视图

quit

(5) 重启成员设备。

reboot chassis *chassis-number*

请将 *chassis-number* 指定为 *member-id* 的值。

1.8.2 配置成员优先级

1. 功能简介

在主设备选举过程中，优先级数值大的成员设备将优先被选举成为主设备。

IRF 形成后，修改成员设备优先级不会触发选举，修改的优先级在下一次选举时生效。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 配置 IRF 中指定成员设备的优先级。

irf member *member-id* **priority** *priority*

缺省情况下，设备的成员优先级均为 1。

1.8.3 配置 IRF 端口

1. 配置限制和指导

同一 IRF 端口绑定的 IRF 物理端口的工作模式必须相同。设备工作在 IRF 模式时，不允许将同一 IRF 端口绑定的 IRF 物理端口配置为不同的工作模式。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 IRF 物理端口视图。

- 进入二层/三层以太网接口视图。

  interface *interface-type interface-number*

- 进入一组接口的批量配置视图。

  interface range { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>

在将一个 IRF 端口与多个物理端口进行绑定时，通过接口批量配置视图可以更快速的完成关闭和开启多个端口的操作。

(3) 关闭接口。

shutdown

接口缺省的状态为开启。

如果允许关闭当前端口，则直接在该接口视图下执行 **shutdown** 命令即可；如果不能关闭该端口，请根据系统提示信息关闭该端口直连的邻居设备上的端口。

(4) 退回系统视图。

quit

(5) 进入 IRF 端口视图。

irf-port *member-id*/*irf-port-number*

(6) 将 IRF 端口和 IRF 物理端口绑定。

port group interface *interface-type interface-number*

缺省情况下，IRF 端口没有和任何 IRF 物理端口绑定。

多次执行该命令，可以将 IRF 端口与多个 IRF 物理端口绑定，以实现 IRF 链路的备份或负载分担，从而提高 IRF 链路的带宽和可靠性。

(7) 退回到系统视图。

quit

(8) 进入 IRF 物理端口视图。

- 进入二层/三层以太网接口视图。

  interface *interface-type interface-number*

- 进入一组接口的批量配置视图。

  interface range { *interface-type interface-number* [**to** *interface-type interface-number*] } &<1-24>

在将一个 IRF 端口与多个物理端口进行绑定时，通过接口批量配置视图可以更快速的完成关闭和开启多个端口的操作。

(9) 打开接口。

undo shutdown

(10) 退回系统视图。

quit

(11) 保存当前配置。

save

激活 IRF 端口会引起 IRF 合并，被选为从设备的成员设备重启。为了避免重启后配置丢失，请在激活 IRF 端口前先将当前配置保存到下次启动配置文件。

(12) 激活 IRF 端口下的配置。

irf-port-configuration active

IRF 物理线缆连接好，并将 IRF 物理端口添加到 IRF 端口后，必须通过该命令手工激活 IRF 端口的配置才能形成 IRF。

1.8.4 快速配置 IRF 基本参数

1. 功能简介

使用本功能，用户可以通过一条命令配置 IRF 的基本参数，包括新成员编号、域编号、成员优先级、绑定物理端口，简化了配置步骤，达到快速配置 IRF 的效果。

在配置该功能时，有两种方式：

- 交互模式：用户输入 **easy-irf**，回车，在交互过程中输入具体参数的值。
- 非交互模式，在输入命令行时直接指定所需参数的值。

两种方式的配置效果相同，如果用户对本功能不熟悉，建议使用交互模式。

2. 配置限制和指导

注意

- 在 IRF 中以设备编号标志设备，配置 IRF 端口和优先级也是根据设备编号来配置的，所以，修改设备成员编号可能导致设备配置发生变化或者丢失，请慎重处理。
- 如果给成员设备指定新的成员编号，该成员设备会立即自动重启，以使新的成员编号生效。

多次使用该功能，修改域编号/优先级/IRF 物理端口时，域编号和优先级的新配置覆盖旧配置，IRF 物理端口的配置会新旧进行叠加。如需删除旧的 IRF 物理端口配置，需要在 IRF 端口视图下，执行 **undo port group interface** 命令。

在交互模式下，为 IRF 端口指定物理端口时，请注意：

- 接口类型和接口编号间不能有空格。
- 不同物理接口之间用英文逗号分隔，逗号前后不能有空格。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 快速配置 IRF。

easy-irf [**member** *member-id* [**renumber** *new-member-id*] **domain** *domain-id* [**priority** *priority*] [**irf-port1** *interface-list1*] [**irf-port2** *interface-list2*]]

若在多成员设备的 IRF 环境中使用该命令，请确保配置的新成员编号与当前 IRF 中的成员编号不冲突。

1.8.5 开启 IRF 合并自动重启功能

1. 功能简介

IRF 合并时，两台 IRF 会遵照角色选举的规则进行竞选，竞选失败方 IRF 的所有成员设备需要重启才能加入获胜方 IRF。如果开启 IRF 合并自动重启功能，则合并过程中的重启由系统自动完成，否则需要用户根据系统提示手工完成重启。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 开启 IRF 合并自动重启功能。

irf auto-merge enable

缺省情况下，IRF 合并自动重启功能处于开启状态。即两台 IRF 合并时，竞选失败方会自动重启。

1.8.6 配置成员设备的描述信息

1. 功能简介

当网络中存在多个 IRF 或者同一 IRF 中存在多台成员设备时可配置成员设备的描述信息进行标识。例如当成员设备的物理位置比较分散（比如在不同楼层甚至不同建筑）时，为了确认成员设备的物理位置，在组建 IRF 时可以将物理位置设置为成员设备的描述信息，以便后期维护。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 配置 IRF 中指定成员设备的描述信息。

irf member *member-id* **description** *text*

缺省情况下，未配置成员设备的描述信息。

1.8.7 配置 IRF 链路的负载分担模式

1. 功能简介

当 IRF 端口与多个 IRF 物理端口绑定时，成员设备之间就会存在多条 IRF 链路。通过改变 IRF 链路负载分担的类型，可以灵活地实现成员设备间流量的负载分担。

用户既可以指定系统按照报文携带的 IP 地址、MAC 地址、报文的入端口号等信息之一或其组合来选择所采用的负载分担模式，也可以指定系统按照报文类型（如二层、IPv4、IPv6 等）自动选择所采用的负载分担模式。

2. 配置限制和指导

用户可以通过全局配置（系统视图下）和端口下（IRF 端口视图下）配置的方式设置 IRF 链路的负载分担模式：

- 系统视图下的配置对所有 IRF 端口生效；
- IRF 端口视图下的配置只对当前 IRF 端口生效；
- IRF 端口会优先采用端口下的配置。如果端口下没有配置，则采用全局配置。

在端口下配置 IRF 链路负载分担模式前，IRF 端口必须至少和一个 IRF 物理端口绑定。否则，端口负载分担模式将配置失败。

3. 全局配置 IRF 链路的负载分担模式

(1) 进入系统视图。

system-view

(2) 配置 IRF 链路的负载分担模式。

irf-port global load-sharing mode { **destination-ip** | **destination-mac** | **ingress-port** | **source-ip** | **source-mac** } *

本命令的缺省情况与设备的接口板型号有关，建议使用缺省值，如需修改请联系技术支持。

多次执行该命令配置不同负载分担模式时，以最新的配置为准。

4. 端口下配置 IRF 链路的负载分担模式

(1) 进入系统视图。

system-view

(2) 进入 IRF 端口视图。

irf-port *member-id*/*irf-port-number*

(3) 配置 IRF 链路的负载分担模式。

irf-port load-sharing mode { **destination-ip** | **destination-mac** | **ingress-port** | **source-ip** | **source-mac** } *

本命令的缺省情况与设备的接口板型号有关，建议使用缺省值，如需修改请联系技术支持。

多次执行该命令配置不同负载分担模式时，以最新的配置为准。

1.8.8 配置 IRF 的桥 MAC 地址

1. 功能简介

桥 MAC 是设备作为网桥与外界通信时使用的 MAC 地址。一些二层协议（例如 LACP）会使用桥 MAC 标识不同设备，所以网络上的桥设备必须具有唯一的桥 MAC。如果网络中存在桥 MAC 相同的设备，则会引起桥 MAC 冲突，从而导致通信故障。IRF 作为一台虚拟设备与外界通信，也具有唯一的桥 MAC，称为 IRF 桥 MAC。

通常情况下，IRF 使用主设备的桥 MAC 作为 IRF 桥 MAC，我们将这台主设备称为 IRF 桥 MAC 拥有者。如果 IRF 桥 MAC 拥有者离开，IRF 继续使用该桥 MAC 的时间可以通过“1.8.8 3. 配置 IRF 的桥 MAC 保留时间”配置。当 IRF 的桥 MAC 保留时间到期后，系统会使用 IRF 中当前主设备的桥 MAC 做 IRF 的桥 MAC。

IRF 合并时，桥 MAC 的处理方式如下：

- IRF 合并时，如果有成员设备的桥 MAC 相同，则它们不能合并为一个 IRF。IRF 的桥 MAC 不受此限制，只要成员设备自身桥 MAC 唯一即可。
- 两台 IRF 合并后，IRF 的桥 MAC 为竞选获胜的一方的桥 MAC。

2. 配置限制和指导

注意

桥 MAC 冲突会引起通信故障，桥 MAC 变化可能导致流量短时间中断，请谨慎配置。

当使用 ARP MAD 和 MSTP 组网或者 ND MAD 和 MSTP 组网时，需要将 IRF 配置为桥 MAC 地址立即改变，即配置 **undo irf mac-address persistent** 命令。

当 IRF 设备上存在跨成员设备的聚合链路时，请不要使用 **undo irf mac-address persistent** 命令配置 IRF 的桥 MAC 立即变化，否则可能会导致流量中断。

3. 配置 IRF 的桥 MAC 保留时间

(1) 进入系统视图。

system-view

(2) 配置 IRF 的桥 MAC 保留时间。请选择其中一项进行配置。

- 配置 IRF 的桥 MAC 永久保留。

  irf mac-address persistent always

  配置 IRF 的桥 MAC 保留时间为 6 分钟。

  irf mac-address persistent timer

- 配置 IRF 的桥 MAC 不保留，立即变化。

  undo irf mac-address persistent

缺省情况下 IRF 桥 MAC 永久保留。

配置 IRF 桥 MAC 保留时间适用于 IRF 桥 MAC 拥有者短时间内离开又回到 IRF 的情况（例如设备重启或者链路临时故障），可以减少不必要的桥 MAC 切换导致的流量中断。

1.8.9 开启启动文件的自动加载功能

1. 功能简介

如果新设备或新主控板加入 IRF，并且新设备/新主控板的软件版本和全局主用主控板的软件版本不一致，则新设备/新主控板不能正常启动。此时：

- 如果没有开启启动文件的自动加载功能，则需要用户手工升级新设备/新主控板后，再将新设备/新主控板加入 IRF。或者在主设备上开启启动文件的自动加载功能，重启新设备/新主控板，让新设备/新主控板重新加入 IRF。
- 如果已经开启了启动文件的自动加载功能，则新设备/新主控板加入 IRF 时，会与全局主用主控板的软件版本号进行比较，如果不一致，则自动从全局主用主控板下载启动文件，然后使用新的系统启动文件重启，重新加入 IRF。如果新下载的启动文件与原有启动文件重名，则原有启动文件会被覆盖。

说明

本功能用于在 IRF 模式下自动保证全局备用主控板和全局主用主控板启动软件包版本的一致性。设备在独立运行模式下时，用户可使用“使能备用主控板启动软件包自动加载功能”来自动保证备用主控板和主用主控板启动软件包版本的一致性。关于“使能备用主控板启动软件包自动加载功能”的详细介绍请参见“基础配置指导”中的“软件升级”。

2. 配置限制和指导

注意

加载启动软件包需要一定时间，在加载期间，请不要插拔或者手工重启处于加载状态的主控板，否则，会导致该主控板加载启动软件包失败而不能启动。用户可打开日志信息显示开关，并根据日志信息的内容来判断加载过程是否开始以及是否结束。

为了能够成功进行自动加载，请确保新加入设备的主控板/新加入主控板的存储介质上有足够的空闲空间用于存放 IRF 的启动文件。如果新加入主控板的存储介质上空闲空间不足，设备将自动删除当前启动文件来再次尝试加载；如果空闲空间仍然不足，该主控板将无法进行自动加载。此时，需要管理员重启该主控板并进入 Boot ROM 菜单，删除一些不重要的文件后，再将主控板重新加入 IRF。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 开启 IRF 系统启动文件的自动加载功能。

irf auto-update enable

缺省情况下，IRF 系统启动文件的自动加载功能处于开启状态。

1.8.10 配置 IRF 链路 down 延迟上报功能

1. 功能简介

该功能用于避免因端口链路层状态在短时间内频繁改变，导致 IRF 分裂/合并的频繁发生。

配置 IRF 链路 down 延迟上报功能后：

- 如果 IRF 链路状态从 up 变为 down，端口不会立即向系统报告链路状态变化。经过一定的时间间隔后，如果 IRF 链路仍然处于 down 状态，端口才向系统报告链路状态的变化，系统再作出相应的处理；
- 如果 IRF 链路状态从 down 变为 up，链路层会立即向系统报告。

2. 配置限制和指导

如果某些协议配置的超时时间小于延迟上报时间（例如 OSPF 等），该协议将超时。此时请适当调整 IRF 链路 down 的延迟上报时间或者该协议的超时时间，使 IRF 链路 down 的延迟上报时间小于协议超时时间，保证协议状态不会发生不必要的切换。

下列情况下，建议将 IRF 链路 down 延迟上报时间配置为 0：

- 对主备倒换速度和 IRF 链路切换速度要求较高时
  - 在 IRF 环境中使用 BFD 功能时
- 在执行关闭 IRF 物理端口或重启 IRF 成员设备的操作之前，请首先将 IRF 链路 down 延迟上报时间配置为 0，待操作完成后再将其恢复为之前的值

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 配置 IRF 链路 down 延迟上报时间。

irf link-delay *interval*

缺省下，IRF 链路 down 延迟上报时间为 4 秒。

1.8.11 拆卸 IRF 物理端口所在的接口模块扩展卡

如果在 IRF 建立后，用户需要拔出 IRF 物理端口所在的接口模块扩展卡，请先拔掉用于 IRF 连接的线缆，或者在 IRF 物理端口视图下执行 **shutdown** 命令关闭该端口，再进行拔出接口模块扩展卡的操作。

1.8.12 更换 IRF 物理端口所在的接口模块扩展卡

如果需要使用不同款型的接口模块扩展卡替换现有接口模块扩展卡进行 IRF 连接，请先解除现有接口模块扩展卡上所有 IRF 物理端口与 IRF 端口的绑定关系，然后拔出现有接口模块扩展卡，安装新接口模块扩展卡后再重新配置新接口模块扩展卡上的端口与 IRF 端口的绑定。

1.9 IRF显示和维护

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后 IRF 的运行情况，通过查看显示信息验证配置的效果。

表1-5 IRF 显示和维护

操作	命令
显示IRF中所有成员设备的相关信息	**display irf**
显示IRF的拓扑信息	**display irf topology**
显示IRF链路信息	**display irf link**
显示所有成员设备上重启以后生效的IRF配置	**display irf configuration**
显示IRF链路的负载分担模式	**display irf-port load-sharing mode** [**irf-port** [*member-id*/*port-number*]]
显示MAD配置信息	**display mad** [**verbose**]

目 录

1 vSystem ····· 1-1
1.1 vSystem 简介 ····· 1-1
1.1.1 vSystem 的应用 ····· 1-1
1.1.2 缺省 vSystem 和非缺省 vSystem ····· 1-1
1.1.3 缺省 vSystem 用户和非缺省 vSystem 用户 ····· 1-2
1.1.4 vSystem 的优点 ····· 1-2
1.2 vSystem 支持模块 ····· 1-2
1.3 vSystem 配置任务简介 ····· 1-4
1.4 创建 vSystem ····· 1-5
1.5 为 vSystem 分配接口和 VLAN 资源 ····· 1-5
1.5.1 为 vSystem 分配接口 ····· 1-5
1.5.2 为 vSystem 分配 VLAN ····· 1-6
1.6 限制 vSystem 的资源使用 ····· 1-6
1.6.1 限制 vSystem 安全策略规则总数 ····· 1-6
1.6.2 限制 vSystem 会话新建速率 ····· 1-6
1.6.3 限制 vSystem 会话并发数 ····· 1-7
1.6.4 限制 vSystem 入方向吞吐量 ····· 1-7
1.7 保存 vSystem 配置 ····· 1-8
1.8 访问和管理 vSystem ····· 1-9
1.9 配置 vSystem 虚拟接口 ····· 1-9
1.10 vSystem 显示和维护 ····· 1-9
1.11 vSystem 典型配置举例 ····· 1-10
1.11.1 基本组网配置举例 ····· 1-10
1.11.2 跨 vSystem 系统流量互访典型配置举例 ····· 1-13
1.11.3 通过 vSystem 实现企业网隔离（非缺省 vSystem 有公网接口） ····· 1-17
1.11.4 通过 vSystem 实现企业网隔离（仅缺省 vSystem 有公网接口） ····· 1-21
1.11.5 通过 vSystem 实现云计算网关 ····· 1-26
1.11.6 Internet 用户通过非缺省 vSystem 的公网接口访问内部服务器 ····· 1-31
1.11.7 Internet 用户通过缺省 vSystem 的公网接口访问内部服务器 ····· 1-36

1 vSystem

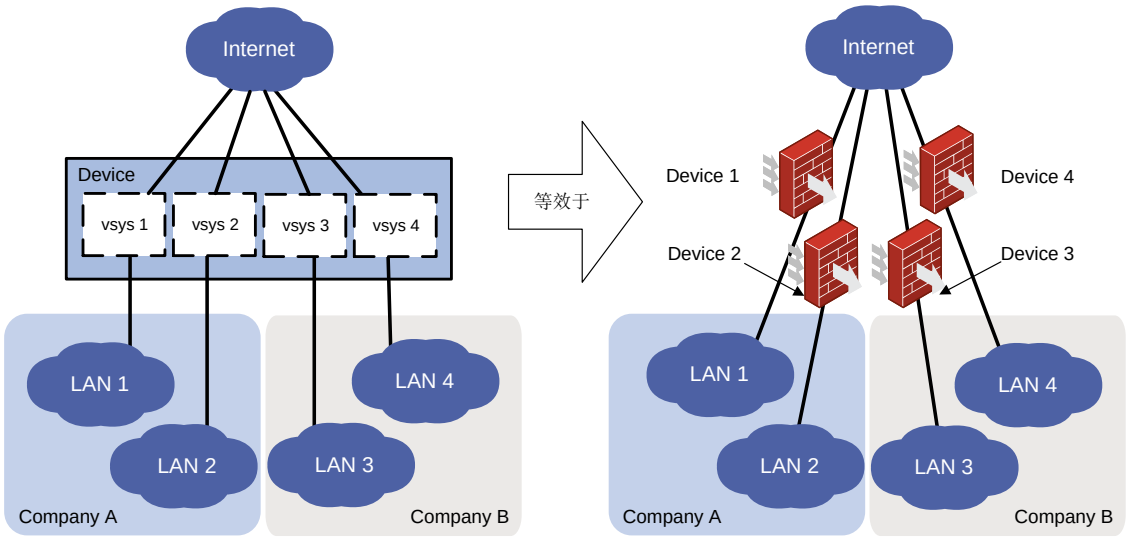
1.1 vSystem简介

vSystem 是一种轻量级的虚拟化技术，能将一台物理设备划分为多台相互独立的逻辑设备。每个 vSystem 相当于一台真实的设备对外服务，拥有独立的接口、VLAN、路由表项、地址范围、策略以及用户/用户组。相比 Context 而言，vSystem 的系统开销更小，因此设备能利用 vSystem 实现更多租户的网络隔离。关于 Context 的详细介绍请参见“虚拟化技术配置指导”中的“Context”。

1.1.1 vSystem 的应用

如图 1-1 所示，LAN 1、LAN 2、LAN 3 和 LAN 4 是四个不同的局域网，其中 LAN 1 和 LAN 2 属于企业 A，LAN 3 和 LAN 4 属于企业 B，它们通过同一台设备 Device 连接到外网。通过虚拟化技术，能将 Device 划分为四台设备使用。具体做法是，在 Device 上创建四个 vSystem（vsys 1、vsys 2、vsys 3 和 vsys 4），分别负责 LAN 1、LAN 2、LAN 3 和 LAN 4 的安全接入。LAN 1、LAN 2、LAN 3 和 LAN 4 的网络管理员可以（也只能）分别登录到自己的 vSystem 进行配置、保存等操作，不会影响其它网络的使用，其效果等同于 LAN 1、LAN 2、LAN 3 和 LAN 4 分别通过各自的设备 Device 1、Device 2、Device 3 和 Device 4 接入 Internet。

图1-1 vSystem 组网示意图



1.1.2 缺省 vSystem 和非缺省 vSystem

设备本身被视作缺省 vSystem，用户在设备内进行配置等同于对缺省 vSystem 进行配置。缺省 vSystem 无需创建、不可删除。缺省 vSystem 拥有设备的所有资源和权限。

用户新创建的 vSystem 被称为非缺省 vSystem，用户可以为非缺省 vSystem 分配接口和 VLAN 资源并指定其它资源限制范围。

未分配给非缺省 vSystem 的接口和 VLAN 资源由缺省 vSystem 使用和管理。非缺省 vSystem 只能使用缺省 vSystem 分配给自己的资源，并在指定的资源限制范围内工作，不能抢占其他 vSystem 的资源。非缺省 vSystem 内不可再创建/删除非缺省 vSystem。

1.1.3 缺省 vSystem 用户和非缺省 vSystem 用户

在缺省 vSystem 内创建的用户被视作缺省 vSystem 用户，缺省 vSystem 用户可以登录到设备内任一非缺省 vSystem 进行业务配置。

在非缺省 vSystem 内创建的用户被称为非缺省 vSystem 用户。非缺省 vSystem 用户只能配置其所属非缺省 vSystem 内的业务，不能登录缺省 vSystem 进行配置。非缺省 vSystem 仅支持部分功能，具体支持情况请参见“1.2 vSystem 支持模块”。

除非特别指明，否则下文中的 vSystem 均指非缺省 vSystem。

1.1.4 vSystem 的优点

vSystem 具备如下优点：

- vSystem 可有效利用设备硬件资源，单一设备可为多个组织提供相互独立的服务，节省能耗和管理成本。
- vSystem 可由统一的缺省 vSystem 用户或相互独立的非缺省 vSystem 用户进行管理，责任清晰、管理灵活。
- 每个 vSystem 拥有独立的策略配置和路由表项，地址空间重叠的用户仍然可以正常通信。
- 每个 vSystem 拥有固定的接口、VLAN 资源以及其它资源限制，业务繁忙的 vSystem 不会对其它 vSystem 造成影响。
- 每个 vSystem 之间的流量相互隔离，安全性高。在需要的时候，vSystem 之间也可以进行安全互访。

1.2 vSystem支持模块

非缺省 vSystem 对各业务模块的支持情况如表 1-1 所示，vSystem 仅支持列入表中的业务模块。完全支持表示 vSystem 支持该模块的所有功能；部分支持表示 vSystem 支持该模块的部分功能，对于部分支持模块的详细支持情况，请参见相应模块的配置指导与命令参考中的 vSystem 相关说明。

表 1-1 中仅是从功能支持层面列出了非缺省 vSystem 中可以支持的所有业务模块，但是每个业务模块在不同产品上的支持情况不同。因此表 1-1 中各业务模块的支持情况，请以设备支持的实际情况为准。


说明

vSystem 下的命令行不支持 **vpn-instance** 参数。

表1-1 vSystem 支持模块列表

模块名称		支持情况
基础配置	CLI	部分支持

	RBAC	部分支持
	配置文件管理	部分支持
	设备管理	部分支持
接口管理	以太网接口	部分支持
	LoopBack接口、NULL接口和InLoopBack接口	部分支持
二层技术-以太网交换	以太网链路聚合	部分支持
	VLAN	部分支持
三层技术-IP业务	ARP	部分支持
	IP地址	部分支持
	IP转发基础	部分支持
	快速转发	部分支持
	邻接表	完全支持
	IPv6基础	部分支持
	IPv6快速转发	部分支持
三层技术-IP路由	IP路由基础	部分支持
	静态路由	部分支持
	OSPF	部分支持
	BGP	部分支持
	IPv6静态路由	部分支持
	OSPFv3	部分支持
IP组播	组播路由与转发	部分支持
	IGMP	部分支持
	PIM	部分支持
	IPv6组播路由与转发	部分支持
	MLD	部分支持
	IPv6 PIM	部分支持
NAT	NAT	部分支持
	AFT	部分支持
ACL和QoS	ACL	部分支持
	时间段	完全支持
VPN	IPsec	部分支持
	IKE	部分支持
安全	安全域	完全支持

	AAA	部分支持
	公钥管理	完全支持
	PKI	部分支持
	SSL	完全支持
	会话管理	部分支持
	连接数限制	完全支持
	对象组	完全支持
	安全策略	部分支持
	攻击检测与防范	部分支持
	ND攻击防御	部分支持
网络管理和监控	系统维护与调试	部分支持
	快速日志输出	部分支持
	Flow日志	部分支持
	信息中心	部分支持
上网行为管理	带宽管理	部分支持
	应用审计与管理	部分支持
负载均衡	服务器负载均衡	部分支持
DPI深度安全	应用层检测引擎	部分支持
	IPS	部分支持
	URL过滤	部分支持
	防病毒	部分支持

1.3 vSystem配置任务简介

vSystem 配置任务如下：

(1) 创建 vSystem

(2) 为 vSystem 分配接口和 VLAN 资源

a. 为 vSystem 分配接口

b. 为 vSystem 分配 VLAN

(3) （可选）限制 vSystem 的资源使用

a. 限制 vSystem 安全策略规则总数

b. 限制 vSystem 会话新建速率

c. 限制 vSystem 会话并发数

d. 限制 vSystem 入方向吞吐量

(4) 保存 vSystem 配置

(5) 访问和管理 vSystem

(6) （可选）配置 vSystem 虚拟接口

1.4 创建vSystem

1. 配置限制和指导

创建 vSystem 时会同时创建一个同名 VPN 实例，因此 vSystem 名称不能与设备内已有的 VPN 实例名称相同，否则将无法创建该名称的 vSystem。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 创建 vSystem，并进入 vSystem 视图。

vsys *vsys-name* [**id** *vsys-id*]

缺省情况下，设备上仅存在缺省 vSystem，名称为 Admin，编号为 1。

(3) （可选）配置 vSystem 的描述信息。

description *text*

缺省情况下，缺省 vSystem 的描述信息为 Default。非缺省 vSystem 没有描述信息。

1.5 为vSystem分配接口和VLAN资源

1.5.1 为 vSystem 分配接口

1. 功能简介

缺省情况下，设备内的所有接口都属于缺省 vSystem，不属于任何非缺省 vSystem。请给非缺省 vSystem 分配接口，它才能使用该接口与网络中的其它设备进行通信。

支持将三层物理接口、三层物理子接口、三层聚合接口、三层聚合子接口、三层冗余接口、三层冗余子接口、Tunnel 接口、LoopBack 接口和 SSLVPN-AC 接口分配给非缺省 vSystem。

将接口分配给某非缺省 vSystem 后，仅该非缺省 vSystem 可以使用该接口。

2. 配置限制和指导

已经与 VPN 实例关联的接口不能分配给 vSystem，反之，已经分配给 vSystem 的接口不能与 VPN 实例关联

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 vSystem 视图。

vsys *vsys-name*

(3) 为 vSystem 分配接口。

allocate interface *interface-type interface-number*

缺省情况下，设备内的所有接口都属于缺省 vSystem，不属于任何非缺省 vSystem。

1.5.2 为 vSystem 分配 VLAN

1. 功能简介

缺省情况下，设备内的所有 VLAN 都属于缺省 vSystem，不属于任何非缺省 vSystem。请给非缺省 vSystem 分配 VLAN，它才能使用该 VLAN 与网络中的其它设备进行通信。

将 VLAN 分配给某非缺省 vSystem 后，其关联的 VLAN 接口会自动分配给该非缺省 vSystem，仅该非缺省 vSystem 可以使用该 VLAN 和 VLAN 接口。

将 VLAN 分配给某非缺省 vSystem 后，该 vSystem 可以使用允许该 VLAN 通过的二层接口。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 vSystem 视图。

vsys *vsys-name*

(3) 为 vSystem 分配 VLAN。

allocate vlan *vlan-id*

缺省情况下，设备内的所有 VLAN 都属于缺省 vSystem，不属于任何非缺省 vSystem。

1.6 限制vSystem的资源使用

1.6.1 限制 vSystem 安全策略规则总数

1. 功能简介

一个 vSystem 内可以配置多个安全策略规则。如果不加限制，会出现大量规则占用过多的内存的情况，影响设备的其它功能正常运行。所以请根据需要为 vSystem 设置安全策略规则总数限制，当安全策略规则总数达到限制值时，该 vSystem 不能继续新增安全策略规则。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 vSystem 视图。

vsys *vsys-name*

(3) 设置 vSystem 的安全策略规则总数限制。

capability security-policy-rule maximum *max-number*

缺省情况下，未对 vSystem 的安全策略规则总数进行限制。

1.6.2 限制 vSystem 会话新建速率

1. 功能简介

为防止一个 vSystem 的会话新建速率过快而导致其他 vSystem 由于处理性能能力不够而无法建立会话，需要限制 vSystem 的会话新建速率，当会话新建速率超过限制值时，超过限制值的会话不会被创建。

2. 配置限制和指导

vSystem 会话新建速率限制对本机流量不生效，例如：FTP、Telnet、SSH、HTTP 和 HTTP 类型的七层负载均衡等业务。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 vSystem 视图。

vsys *vsys-name*

(3) 设置 vSystem 的会话新建速率限制。

capability session rate *max-value*

缺省情况下，未对 vSystem 的会话新建速率进行限制。

1.6.3 限制 vSystem 会话并发数

1. 功能简介

为防止一个 vSystem 建立了太多会话而导致其他 vSystem 的会话由于内存不够而无法建立，需要限制 vSystem 建立会话的数量，当会话总数达到限制值时，该 vSystem 不能继续新建会话。已经创建的会话不会被删除，直到已建立的会话通过老化机制使得会话总数低于配置的会话并发数限制后，该 vSystem 才允许新建会话。

2. 配置限制和指导

vSystem 会话并发数限制对本机流量不生效，例如：FTP、Telnet、SSH、HTTP 和 HTTP 类型的七层负载均衡等业务。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 vSystem 视图。

vsys *vsys-name*

(3) 设置 vSystem 的会话并发数限制。

capability session maximum *max-number*

缺省情况下，未对 vSystem 的会话并发数进行限制。

1.6.4 限制 vSystem 入方向吞吐量

1. 功能简介

为防止因个别 vSystem 的带宽过大而导致其他 vSystem 由于缺少带宽而无法转发流量，设备需要对 vSystem 的入方向吞吐量进行限制，vSystem 仅能以小于或等于吞吐量限制的带宽转发报文。

除此之外，还可以针对 vSystem 的入方向吞吐量限制开启吞吐量告警功能以及吞吐量限速丢包日志功能。

- 入方向吞吐量告警功能：开启此功能并设置告警阈值后，当 vSystem 的入方向吞吐量与入方向吞吐量限制值的比值超过了所设置的告警阈值，设备会生成告警日志；之后，当 vSystem 的入方向吞吐量与入方向吞吐量限制值的比值恢复到告警阈值以下，设备会生成恢复日志。
- 入方向吞吐量限速丢包日志功能：开启此功能后，当 vSystem 的入方向吞吐量达到入方向吞吐量限制值，设备会将超出限制值的报文丢弃，并对丢弃的报文生成日志信息；之后，如果该 vSystem 的入方向吞吐量降低到入方向吞吐量限制值以下，设备会生成恢复日志。

上面生成的日志信息将会被输出到信息中心模块处理，信息中心模块的配置将决定日志信息的发送规则和发送方向。有关信息中心的详细介绍，请参见“网络管理和监控配置指导”中的“信息中心”。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 vSystem 视图。

vsys *vsys-name*

(3) 设置 vSystem 入方向的吞吐量限制。

capability throughput { kbps | pps } *threshold*

缺省情况下，未对 vSystem 的吞吐量进行限制。

(4) （可选）开启 vSystem 入方向吞吐量告警功能并设置告警阈值。

vsys-capability throughput alarm enable alarm-threshold *alarm-threshold*

缺省情况下，vSystem 入方向吞吐量告警功能处于关闭状态。

(5) （可选）开启 vSystem 入方向吞吐量限速丢包日志功能。

vsys-capability throughput drop-logging enable

缺省情况下，vSystem 入方向吞吐量限速丢包日志功能处于关闭状态。

1.7 保存vSystem配置

1. 功能简介

本功能可将指定 vSystem 的配置保存到指定文本文件中。如果指定的文件名不存在，系统会先创建该文件，再执行保存操作，否则将覆盖同名文件。

如果不指定文件路径或者指定的文件路径是设备的下次启动配置文件路径，本功能会将指定的 vSystem 的配置保存到设备的下次启动配置文件的相应区段中，设备重启后该 vSystem 将按保存的配置恢复。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 保存指定 vSystem 的当前配置。

save vsys *vsys-name* [*file-url*]

1.8 访问和管理vSystem

1. 功能简介

缺省 vSystem 用户可以使用 **switchto vsys** 命令，通过设备和 vSystem 的内部连接登录 vSystem 进行配置和管理。

非缺省 vSystem 用户可以使用 vSystem 上的接口 IP 地址进行 Telnet/SSH 登录。利用此方式登录，用户名需加上后缀“@@*vsysname*”，其中 *vsysname* 是 vSystem 的名称。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 登录指定 vSystem。

switchto vsys *vsys-name*

缺省 vSystem 用户登录 vSystem 后，可以在 vSystem 的用户视图执行 **quit** 命令来退出登录。此时，命令视图将从当前 vSystem 的用户视图返回到缺省 vSystem 的系统视图。

1.9 配置vSystem虚拟接口

1. 功能简介

vSystem 虚拟接口用于非缺省 vSystem 之间的通信。vSystem 虚拟接口在用户创建非缺省 vSystem 时由设备自动创建。

每个 vSystem 只会创建一个 vSystem 虚拟接口，此接口不支持手工创建。可通过执行 **display interface vsys-interface** 命令查看 vSystem 虚拟接口详细情况。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 vSystem 虚拟接口视图。

interface vsys-interface *interface-number*

(3) （可选）设置接口的描述信息

description *text*

(4) （可选）恢复接口的缺省配置

default

1.10 vSystem显示和维护

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后 vSystem 的运行情况，通过查看显示信息验证配置的效果。

在用户视图下，执行 **reset** 命令可以清除 vSystem 的统计信息。

表1-2 缺省 vSystem 上可执行的显示和维护

操作	命令
显示vSystem虚拟接口的相关信息	**display interface** [**vsys-interface** [*interface-number*]] [**brief** [**description**]]
显示vSystem内吞吐量资源的使用情况	（独立运行模式） **display vsys-capability throughput** [**name** *vsys-name*] [**slot** *slot-number* **cpu** *cpu-number*] （IRF模式） **display vsys-capability throughput** [**name** *vsys-name*] [**chassis** *chassis-number* **slot** *slot-number* **cpu** *cpu-number*]
显示vSystem的相关信息	**display vsys** [**name** *vsys-name*]
显示vSystem的接口列表	**display vsys** [**name** *vsys-name*] **interface**
显示vSystem的VLAN列表	**display vsys** [**name** *vsys-name*] **vlan**
清除vSystem虚拟接口的统计信息	**reset counters interface** [**vsys-interface** [*interface-number*]]

表1-3 非缺省 vSystem 上可执行的显示和维护

操作	命令
显示vSystem虚拟接口的相关信息	**display interface** [**vsys-interface** [*interface-number*]] [**brief** [**description**]]
清除vSystem虚拟接口的统计信息	**reset counters interface** [**vsys-interface** [*interface-number*]]

1.11 vSystem典型配置举例

1.11.1 基本组网配置举例

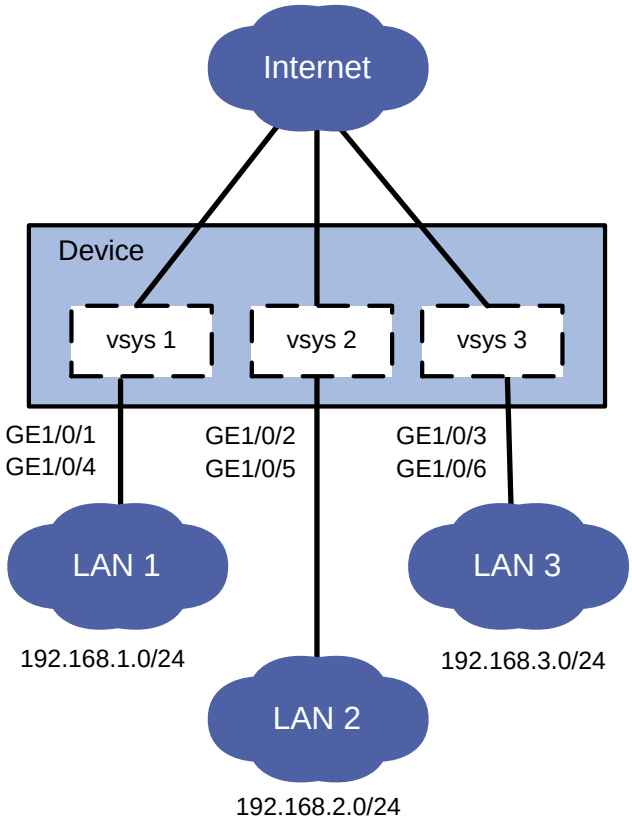
1. 组网需求

LAN 1（192.168.1.0/24 网段）、LAN 2（192.168.2.0/24 网段）、LAN 3（192.168.3.0/24 网段）分别属于公司 A、公司 B、公司 C，现需要对各公司的网络进行独立的安全防护，具体需求如下：

- 将设备 Device 虚拟成三台独立的 Device，并分给三个不同的用户网络进行安全防护。要求在用户侧看来，各自的接入设备是独享的。
- 将接口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/4 分配给 A 公司，将接口 GigabitEthernet1/0/2 和 GigabitEthernet1/0/5 分配给 B 公司，将接口 GigabitEthernet1/0/3 和 GigabitEthernet1/0/6 分配给 C 公司。

2. 组网图

图1-2 配置 vSystem 组网图



3. 配置步骤

(1) 创建并配置 vSystem vsys1，供公司 A 使用

创建 vsys1，设置描述信息。

```
<Device> system-view
[Device] vsys vsys1
[Device-vsys-2-vsys1] description vsys-1
```

将接口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/4 分配给 vsys1。

```
[Device-vsys-2-vsys1] allocate interface gigabitethernet 1/0/1
Some configurations on the interface are removed.
[Device-vsys-2-vsys1] allocate interface gigabitethernet 1/0/4
Some configurations on the interface are removed.
[Device-vsys-2-vsys1] quit
```

登录 vsys1。

```
[Device] switchto vsys vsys1
<Device-vsys1> system-view
```

配置 Telnet 功能，保证管理用户可以正常登录设备，具体配置步骤请参考“基础配置指导”中的“登录设备”。

配置接口 GigabitEthernet1/0/1 的 IP 地址为 192.168.1.251，供公司 A 的管理用户远程登录。

```
[Device-vsys1] interface gigabitethernet 1/0/1
[Device-vsys1-GigabitEthernet1/0/1] ip address 192.168.1.251 24
```

从 vsys1 返回缺省 vSystem。

```
[Device-vsys1-GigabitEthernet1/0/1] return
<Device-vsys1> quit
[Device]
```

(2) 创建并配置 vSystem vsys2，供公司 B 使用

创建 vsys2，设置描述信息。

```
[Device] vsys vsys2
[Device-vsys-3-vsys2] description vsys-2
```

将接口 GigabitEthernet1/0/2 和 GigabitEthernet1/0/5 分配给 vsys2。

```
[Device-vsys-3-vsys2] allocate interface gigabitethernet 1/0/2
Some configurations on the interface are removed.
[Device-vsys-3-vsys2] allocate interface gigabitethernet 1/0/5
Some configurations on the interface are removed.
[Device-vsys-3-vsys2] quit
```

登录 vsys2。

```
[Device] switchto vsys vsys2
<Device-vsys2> system-view
```

配置 Telnet 功能，保证管理用户可以正常登录设备，具体配置步骤请参考“基础配置指导”中的“登录设备”。

配置接口 GigabitEthernet1/0/2 的 IP 地址为 192.168.2.251，供公司 B 的管理用户远程登录。

```
[Device-vsys2] interface gigabitethernet 1/0/2
[Device-vsys2-GigabitEthernet1/0/2] ip address 192.168.2.251 24
```

从 vsys2 返回缺省 vSystem。

```
[Device-vsys2-GigabitEthernet1/0/2] return
<Device-vsys2> quit
[Device]
```

(3) 创建并配置 vSystem vsys3，供公司 C 使用

创建 vsys3，设置描述信息

```
[Device] vsys vsys3
[Device-vsys-4-vsys3] description vsys-3
```

将接口 GigabitEthernet1/0/3 和 GigabitEthernet1/0/6 分配给 vsys3。

```
[Device-vsys-4-vsys3] allocate interface gigabitethernet 1/0/3
```

```
Some configurations on the interface are removed.
[Device-vsys-4-vsys3] allocate interface gigabitethernet 1/0/6
Some configurations on the interface are removed.
[Device-vsys-4-vsys3] quit
```

登录 vsys3。

```
[Device] switchto vsys vsys3
<Device-vsys2> system-view
```

配置 Telnet 功能，保证管理用户可以正常登录设备，具体配置步骤请参考“基础配置指导”中的“登录设备”。

配置接口 GigabitEthernet1/0/3 的 IP 地址为 192.168.3.251，供公司 C 的管理用户远程登录。

```
[Device-vsys3] interface gigabitethernet 1/0/3
[Device-vsys3-GigabitEthernet1/0/3] ip address 192.168.3.251 24
```

从 vsys3 返回缺省 vSystem。

```
[Device-vsys3-GigabitEthernet1/0/3] return
<Device-vsys3> quit
[Device]
```

4. 验证配置

在 Device 上查看所配 vSystem 是否存在并且运转正常。（此时，Device 上应该有四台处于正常工作 active 状态的 vSystem）

```
[Device] display vsys

ID       Name          Status           Description
1        Admin         Active           Default
2        vsys1         Active           vsys-1
3        vsys2         Active           vsys-2
4        vsys3         Active           vsys-3
```

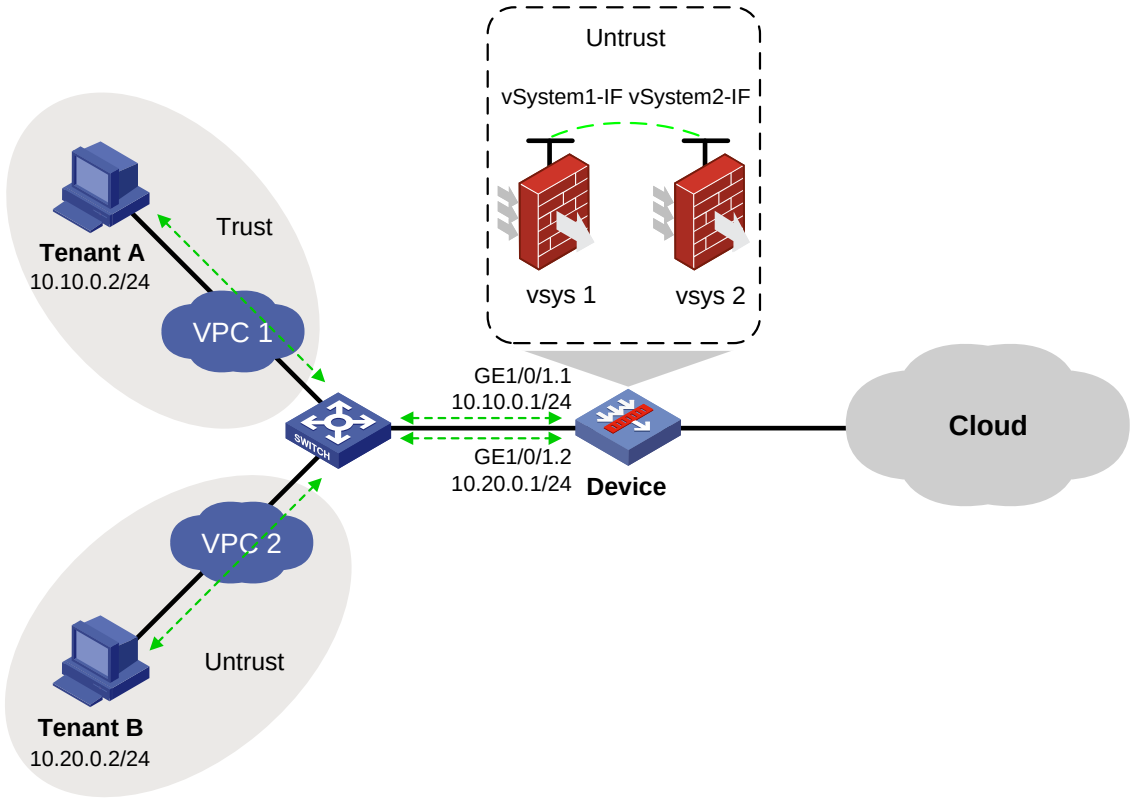
1.11.2 跨 vSystem 系统流量互访典型配置举例

1. 组网需求

设备上的非缺省 vSystem 在对 VPC 租户访问公有云的流量进行安全防护的同时也可以使不同 VPC 之间的流量互通。

2. 组网图

图1-3 跨 vSystem 系统流量互访组网图



3. 配置步骤

(1) 创建子接口 GigabitEthernet1/0/1.1 和 GigabitEthernet1/0/1.2。

创建子接口 GigabitEthernet1/0/1.1。

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1.1
```

配置子接口 GigabitEthernet1/0/1.1 能够终结最外层 VLAN ID 为 10 的 VLAN 报文。

```
[Device-GigabitEthernet1/0/1.1] vlan-type dot1q vid 10
[Device-GigabitEthernet1/0/1.1] quit
```

创建子接口 GigabitEthernet1/0/1.2。

```
[Device] interface gigabitethernet 1/0/1.2
```

配置子接口 GigabitEthernet1/0/1.2 能够终结最外层 VLAN ID 为 20 的 VLAN 报文。

```
[Device-GigabitEthernet1/0/1.2] vlan-type dot1q vid 20
[Device-GigabitEthernet1/0/1.2] quit
```

(2) 创建并配置 vSystem vsys1，供公司 VPC1 使用

创建 vsys1，设置描述信息。

```
[Device] vsys vsys1
[Device-vsys-2-vsys1] description vsys1
```

将子接口 GigabitEthernet1/0/1.1 分配给 vsys1。

```
[Device-vsys-2-vsys1] allocate interface gigabitethernet 1/0/1.1
[Device-vsys-2-vsys1] quit
```

(3) 创建并配置 vSystem vsys2，供公司 VPC2 使用

创建 vsys2，设置描述信息。

```
[Device] vsys vsys2
[Device-vsys-3-vsys2] description vsys2
```

将子接口 GigabitEthernet1/0/1.2 分配给 vsys2。

```
[Device-vsys-3-vsys2] allocate interface gigabitethernet 1/0/1.2
[Device-vsys-3-vsys2] quit
```

(4) 配置静态路由，保证路由可达。

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由使 VPC 1 和 VPC 2 之间路由可达，具体配置步骤如下。

```
[Device] ip route-static vpn-instance vsys1 10.20.0.0 24 vpn-instance vsys2
[Device] ip route-static vpn-instance vsys2 10.10.0.0 24 vpn-instance vsys1
```

(5) 在 vsys1 上配置安全域。

将以太网子接口和 vSystem 虚拟接口加入安全域。

```
[Device] switchto vsys vsys1
<Device-vsys1> system-view
[Device-vsys1] security-zone name trust
[Device-vsys1-security-zone-Trust] import interface gigabitethernet 1/0/1.1
[Device-vsys1-security-zone-Trust] quit
[Device-vsys1] security-zone name untrust
[Device-vsys1-security-zone-Untrust] import interface vsys-interface2
[Device-vsys1-security-zone-Untrust] quit
```

(6) 在 vsys1 上配置安全策略保证 VPC 之间的流量互通。

```
[Device-vsys1] security-policy ip
[Device-vsys1-security-policy-ip] rule 0 name vpc1
[Device-vsys1-security-policy-ip-0-vpc1] action pass
[Device-vsys1-security-policy-ip-0-vpc1] source-zone trust
[Device-vsys1-security-policy-ip-0-vpc1] destination-zone untrust
[Device-vsys1-security-policy-ip-0-vpc1] quit
[Device-vsys1-security-policy-ip] rule 1 name vpc2
[Device-vsys1-security-policy-ip-1-vpc2] action pass
[Device-vsys1-security-policy-ip-1-vpc2] source-zone untrust
[Device-vsys1-security-policy-ip-1-vpc2] destination-zone trust
[Device-vsys1-security-policy-ip-1-vpc2] return
```

```
<Device-vsys1> quit
[Device]
```

(7) 在 vsys2 上配置安全域。

将以太网子接口和 vSystem 虚拟接口加入安全域。

```
[Device] switchto vsys vsys2
<Device-vsys2> system-view
[Device-vsys2] security-zone name trust
[Device-vsys2-security-zone-Trust] import interface gigabitethernet 1/0/1.2
[Device-vsys2-security-zone-Trust] quit
[Device-vsys2] security-zone name untrust
[Device-vsys2-security-zone-Untrust] import interface vsys-interface3
[Device-vsys2-security-zone-Untrust] quit
```

(8) 在 vsys2 上配置安全策略保证 VPC 之间的流量互通。

```
[Device-vsys2] security-policy ip
[Device-vsys2-security-policy-ip] rule 0 name vpc1
[Device-vsys2-security-policy-ip-0-vpc1] action pass
[Device-vsys2-security-policy-ip-0-vpc1] source-zone trust
[Device-vsys2-security-policy-ip-0-vpc1] destination-zone untrust
[Device-vsys2-security-policy-ip-0-vpc1] quit
[Device-vsys2-security-policy-ip] rule 1 name vpc2
[Device-vsys2-security-policy-ip-1-vpc2] action pass
[Device-vsys2-security-policy-ip-1-vpc2] source-zone untrust
[Device-vsys2-security-policy-ip-1-vpc2] destination-zone trust
[Device-vsys2-security-policy-ip-1-vpc2] quit
[Device-vsys2-security-policy-ip] quit
```

4. 验证配置

在租户 A 上可以 Ping 通租户 B。

```
C:\> ping 10.20.0.2

Pinging 10.20.0.2 with 32 bytes of data:

Reply from 10.20.0.2: bytes=32 time=19ms TTL=254
Reply from 10.20.0.2: bytes=32 time<1ms TTL=254
Reply from 10.20.0.2: bytes=32 time<1ms TTL=254
Reply from 10.20.0.2: bytes=32 time<1ms TTL=254

Ping statistics for 10.20.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0  (0% loss),
```

```
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 19ms, Average = 4ms
```

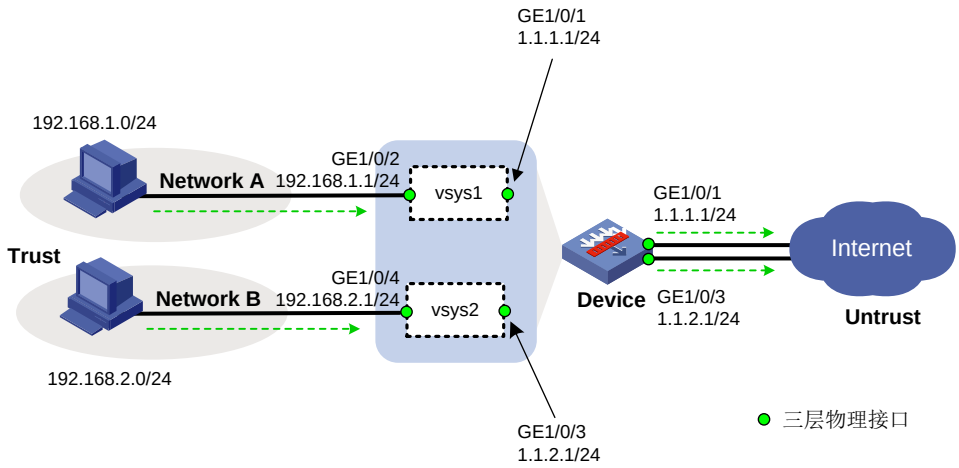
1.11.3 通过 vSystem 实现企业网隔离（非缺省 vSystem 有公网接口）

1. 组网需求

某公司网络出口部署了一台 Device 作为安全网关，其内部划分为研发部门网络 A 和非研发部门网络 B，网络 A 和网络 B 之间相互隔离，其用户不能互访。此外，网络 A 中只有部分用户（192.168.1.128/25 网段）可以通过独立的公网接口访问 Internet，而网络 B 中的所有用户都可以通过独立的公网接口访问 Internet。

2. 组网图

图1-4 通过 vSystem 实现企业网隔离（非缺省 vSystem 有公网接口）



3. 配置 vSystem vsys1

(1) 创建 vSystem vsys1，并为其分配接口

创建 vsys1，设置描述信息。

```
<Device> system-view
[Device] vsys vsys1
[Device-vsys-2-vsys1] description vsys-1
```

将接口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/2 分配给 vsys1。

```
[Device-vsys-2-vsys1] allocate interface gigabitethernet 1/0/1
Some configurations on the interface are removed.
[Device-vsys-2-vsys1] allocate interface gigabitethernet 1/0/2
Some configurations on the interface are removed.
[Device-vsys-2-vsys1] quit
```

(2) 登录 vsys1，配置接口 IP 地址及其所属安全域

登录 vsys1。

```
[Device] switchto vsys vsys1
```

```
<Device-vsys1> system-view
```

根据组网图中规划的信息，配置各接口的 IP 地址，具体配置步骤如下。

```
[Device-vsys1] interface gigabitethernet 1/0/1
[Device-vsys1-GigabitEthernet1/0/1] ip address 1.1.1.1 24
[Device-vsys1-GigabitEthernet1/0/1] quit
[Device-vsys1] interface gigabitethernet 1/0/2
[Device-vsys1-GigabitEthernet1/0/2] ip address 192.168.1.1 24
[Device-vsys1-GigabitEthernet1/0/2] quit
```

根据组网图中规划的信息，将接口加入对应的安全域，具体配置步骤如下。

```
[Device-vsys1] security-zone name trust
[Device-vsys1-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-vsys1-security-zone-Trust] quit
[Device-vsys1] security-zone name untrust
[Device-vsys1-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Device-vsys1-security-zone-Untrust] quit
```

(3) 配置静态路由

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由，本举例假设 vsys1 内用户访问 Internet 的下一跳 IP 地址为 1.1.1.2，实际使用中请以具体组网情况为准，具体配置步骤如下。

```
[Device-vsys1] ip route-static 0.0.0.0 0 1.1.1.2
```

(4) 配置安全策略

配置名称为 trust-untrust 的安全策略，保证 192.168.1.128/25 网段的用户可以访问 Internet，具体配置步骤如下。

```
[Device-vsys1] security-policy ip
[Device-vsys1-security-policy-ip] rule name trust-untrust
[Device-vsys1-security-policy-ip-1-trust-untrust] source-zone trust
[Device-vsys1-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-vsys1-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.1.128 25
[Device-vsys1-security-policy-ip-1-trust-untrust] action pass
[Device-vsys1-security-policy-ip-1-trust-untrust] quit
[Device-vsys1-security-policy-ip] quit
```

(5) 配置 NAT 功能

配置 ACL 2000，仅允许对来自 192.168.1.128/25 网段的报文进行地址转换。

```
[Device-vsys1] acl basic 2000
[Device-vsys1-acl-ipv4-basic-2000] rule permit source 192.168.1.128 0.0.0.127
[Device-vsys1-acl-ipv4-basic-2000] quit
```

在接口 GigabitEthernet1/0/1 上配置 Easy IP 方式的出方向动态地址转换，使得匹配指定 ACL 的内网用户访问 Internet 的报文可以使用接口 GigabitEthernet1/0/1 的 IP 地址进行源地址转换。

```
[Device-vsys1] interface gigabitethernet 1/0/1
[Device-vsys1-GigabitEthernet1/0/1] nat outbound 2000
```

从 vsys1 返回缺省 vSystem。

```
[Device-vsys1-GigabitEthernet1/0/1] return
<Device-vsys1> quit
[Device]
```

4. 配置 vSystem vsys2

(1) 创建 vSystem vsys2，并为其分配接口

创建 vsys2，设置描述信息。

```
[Device] vsys vsys2
[Device-vsys-3-vsys2] description vsys-2
```

将接口 GigabitEthernet1/0/3 和 GigabitEthernet1/0/4 分配给 vsys2。

```
[Device-vsys-3-vsys2] allocate interface gigabitethernet 1/0/3
Some configurations on the interface are removed.
[Device-vsys-3-vsys2] allocate interface gigabitethernet 1/0/4
Some configurations on the interface are removed.
[Device-vsys-3-vsys2] quit
```

(2) 登录 vsys2，配置接口 IP 地址及其所属安全域

登录 vsys2。

```
[Device] switchto vsys vsys2
<Device-vsys2> system-view
```

根据组网图中规划的信息，配置各接口的 IP 地址，具体配置步骤如下。

```
[Device-vsys2] interface gigabitethernet 1/0/3
[Device-vsys2-GigabitEthernet1/0/3] ip address 1.1.2.1 24
[Device-vsys2-GigabitEthernet1/0/3] quit
[Device-vsys2] interface gigabitethernet 1/0/4
[Device-vsys2-GigabitEthernet1/0/4] ip address 192.168.2.1 24
[Device-vsys2-GigabitEthernet1/0/4] quit
```

根据组网图中规划的信息，将接口加入对应的安全域，具体配置步骤如下。

```
[Device-vsys2] security-zone name trust
[Device-vsys2-security-zone-Trust] import interface gigabitethernet 1/0/4
[Device-vsys2-security-zone-Trust] quit
[Device-vsys2] security-zone name untrust
[Device-vsys2-security-zone-Untrust] import interface gigabitethernet 1/0/3
[Device-vsys2-security-zone-Untrust] quit
```

(3) 配置静态路由

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由，本举例假设 vsys2 内用户访问 Internet 的下一跳 IP 地址为 1.1.2.2，实际使用中请以具体组网情况为准，具体配置步骤如下。

```
[Device-vsys2] ip route-static 0.0.0.0 0 1.1.2.2
```

(4) 配置安全策略

配置名称为 trust-untrust 的安全策略，保证 192.168.2.0/24 网段的用户可以访问 Internet，具体配置步骤如下。

```
[Device-vsys2] security-policy ip
[Device-vsys2-security-policy-ip] rule name trust-untrust
[Device-vsys2-security-policy-ip-1-trust-untrust] source-zone trust
[Device-vsys2-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-vsys2-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.2.0 24
[Device-vsys2-security-policy-ip-1-trust-untrust] action pass
[Device-vsys2-security-policy-ip-1-trust-untrust] quit
[Device-vsys2-security-policy-ip] quit
```

(5) 配置 NAT 功能

配置 ACL 2000，仅允许对来自 192.168.2.0/24 网段的报文进行地址转换。

```
[Device-vsys2] acl basic 2000
[Device-vsys2-acl-ipv4-basic-2000] rule permit source 192.168.2.0 0.0.0.255
[Device-vsys2-acl-ipv4-basic-2000] quit
```

在接口 GigabitEthernet1/0/3 上配置 Easy IP 方式的出方向动态地址转换，使得匹配指定 ACL 的内网用户访问 Internet 的报文可以使用接口 GigabitEthernet1/0/3 的 IP 地址进行源地址转换。

```
[Device-vsys2] interface gigabitethernet 1/0/3
[Device-vsys2-GigabitEthernet1/0/3] nat outbound 2000
```

从 vsys2 返回缺省 vSystem。

```
[Device-vsys2-GigabitEthernet1/0/3] return
<Device-vsys2> quit
[Device]
```

5. 验证配置

(1) 在 Device 上查看所配 vSystem 是否存在并且运转正常。（此时，Device 上应该有三台处于正常工作 active 状态的 vSystem）

```
[Device] display vsys

ID       Name          Status            Description
1        Admin         Active            Default
2        vsys1         Active            vsys-1
```

```
3        vsys2         Active            vsys-2
```

(2) 位于 192.168.1.128/25 网段的用户可以成功访问 Internet。

```
C:\> ping 3.3.3.3

Pinging 3.3.3.3 with 32 bytes of data:

Reply from 3.3.3.3: bytes=32 time=51ms TTL=255

Reply from 3.3.3.3: bytes=32 time=44ms TTL=255

Reply from 3.3.3.3: bytes=32 time=1ms TTL=255

Reply from 3.3.3.3: bytes=32 time=1ms TTL=255

Ping statistics for 3.3.3.3:

    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

    Minimum = 1ms, Maximum = 51ms, Average = 24ms
```

(3) 位于 192.168.2./24 网段的用户可以成功访问 Internet。

```
C:\> ping 3.3.3.3

Pinging 3.3.3.3 with 32 bytes of data:

Reply from 3.3.3.3: bytes=32 time=25ms TTL=255

Reply from 3.3.3.3: bytes=32 time=36ms TTL=255

Reply from 3.3.3.3: bytes=32 time=1ms TTL=255

Reply from 3.3.3.3: bytes=32 time=1ms TTL=255

Ping statistics for 3.3.3.3:

    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

    Minimum = 1ms, Maximum = 36ms, Average = 16ms
```

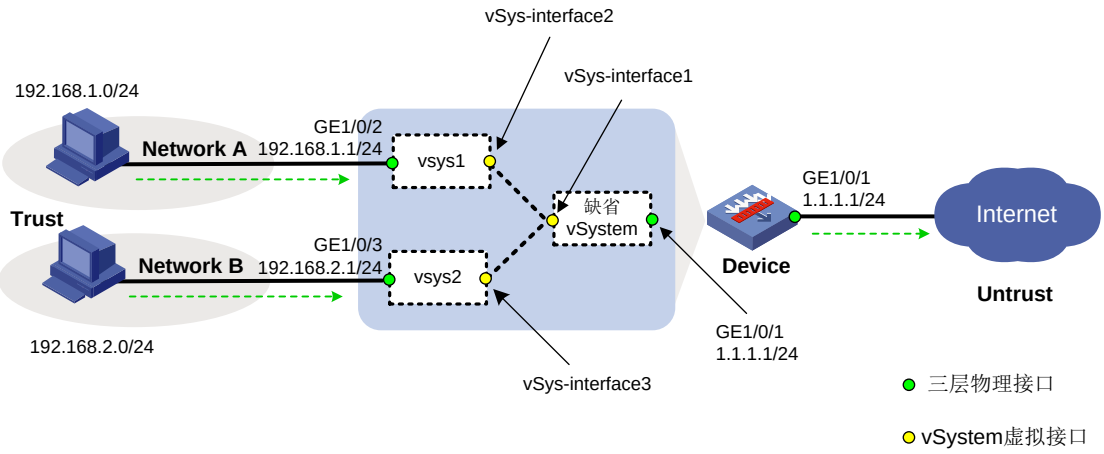
1.11.4 通过 vSystem 实现企业网隔离（仅缺省 vSystem 有公网接口）

1. 组网需求

某公司网络出口部署了一台 Device 作为安全网关，其内部划分为研发部门网络 A 和非研发部门网络 B，网络 A 和网络 B 之间相互隔离，其用户不能互访。此外，网络 A 中只有部分用户（192.168.1.128/25 网段）可以访问 Internet，而网络 B 中的所有用户都可以访问 Internet。

2. 组网图

图1-5 通过 vSystem 实现企业网隔离（仅缺省 vSystem 有公网接口）组网图



3. 配置缺省 vSystem

(1) 创建 vSystem，并为其分配接口

创建 vsys1，设置描述信息。

```
<Device> system-view
[Device] vsys vsys1
[Device-vsys-2-vsys1] description vsys-1
```

将接口 GigabitEthernet1/0/2 分配给 vsys1。

```
[Device-vsys-2-vsys1] allocate interface gigabitethernet 1/0/2
Some configurations on the interface are removed.
[Device-vsys-2-vsys1] quit
```

创建 vsys2，设置描述信息。

```
[Device] vsys vsys2
[Device-vsys-3-vsys2] description vsys-2
```

将接口 GigabitEthernet1/0/3 分配给 vsys2。

```
[Device-vsys-3-vsys2] allocate interface gigabitethernet 1/0/3
Some configurations on the interface are removed.
[Device-vsys-3-vsys2] quit
```

(2) 配置接口 IP 地址及其所属安全域

根据组网图中规划的信息，配置各接口的 IP 地址，具体配置步骤如下。

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 1.1.1.1 24
[Device-GigabitEthernet1/0/1] quit
```

根据组网图中规划的信息，将接口加入对应的安全域，具体配置步骤如下。

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface vsys-interface 1
```

```
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Device-security-zone-Untrust] quit
```

(3) 配置静态路由

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由，本举例假设内网用户通过缺省 vSystem 访问 Internet 的下一跳 IP 地址为 1.1.1.2，实际使用中请以具体组网情况为准，具体配置步骤如下。

```
[Device] ip route-static 0.0.0.0 0 1.1.1.2
[Device] ip route-static 192.168.1.0 24 vpn-instance vsys1
[Device] ip route-static 192.168.2.0 24 vpn-instance vsys2
```

(4) 配置安全策略

配置名称为 trust-untrust 的安全策略，保证内网用户可以访问 Internet，具体配置步骤如下。

```
[Device] security-policy ip
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-1-trust-untrust] source-zone trust
[Device-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.1.128 25
[Device-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.2.0 24
[Device-security-policy-ip-1-trust-untrust] action pass
[Device-security-policy-ip-1-trust-untrust] quit
[Device-security-policy-ip] quit
```

(5) 配置 NAT 功能

配置 ACL 2000，仅允许对来自 192.168.1.128/25 网段和 192.168.2.0/24 网段的报文进行地址转换。

```
[Device] acl basic 2000
[Device-acl-ipv4-basic-2000] rule permit source 192.168.1.128 0.0.0.127
[Device-acl-ipv4-basic-2000] rule permit source 192.168.2.0 0.0.0.255
[Device-acl-ipv4-basic-2000] quit
```

在接口 GigabitEthernet1/0/1 上配置 Easy IP 方式的出方向动态地址转换，使得匹配指定 ACL 的内网用户访问 Internet 的报文可以使用接口 GigabitEthernet1/0/1 的 IP 地址进行源地址转换。

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] nat outbound 2000
[Device-GigabitEthernet1/0/1] quit
```

4. 配置 vSystem vsys1

(1) 登录 vsys1，配置接口 IP 地址及其所属安全域

登录 vsys1。

```
[Device] switchto vsys vsys1
<Device-vsys1> system-view
```

根据组网图中规划的信息，配置各接口的 IP 地址，具体配置步骤如下。

```
[Device-vsys1] interface gigabitethernet 1/0/2
[Device-vsys1-GigabitEthernet1/0/2] ip address 192.168.1.1 24
[Device-vsys1-GigabitEthernet1/0/2] quit
```

根据组网图中规划的信息，将接口加入对应的安全域，具体配置步骤如下。

```
[Device-vsys1] security-zone name trust
[Device-vsys1-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-vsys1-security-zone-Trust] quit
[Device-vsys1] security-zone name untrust
[Device-vsys1-security-zone-Untrust] import interface vsys-interface 2
[Device-vsys1-security-zone-Untrust] quit
```

(2) 配置静态路由

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由，本举例假设 vsys1 内用户访问 Internet 的下一跳为缺省 vSystem，实际使用中请以具体组网情况为准，具体配置步骤如下。

```
[Device-vsys1] ip route-static 0.0.0.0 0 public
```

(3) 配置安全策略

配置名称为 trust-untrust 的安全策略，保证 192.168.1.128/25 网段的用户可以访问 Internet，具体配置步骤如下。

```
[Device-vsys1] security-policy ip
[Device-vsys1-security-policy-ip] rule name trust-untrust
[Device-vsys1-security-policy-ip-1-trust-untrust] source-zone trust
[Device-vsys1-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-vsys1-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.1.128 25
[Device-vsys1-security-policy-ip-1-trust-untrust] action pass
```

从 vsys1 返回缺省 vSystem。

```
[Device-vsys1-security-policy-ip-1-trust-untrust] return
<Device-vsys1> quit
[Device]
```

5. 配置 vSystem vsys2

(1) 登录 vsys2，配置接口 IP 地址及其所属安全域

登录 vsys2。

```
[Device] switchto vsys vsys2
<Device-vsys2> system-view
```

根据组网图中规划的信息，配置各接口的 IP 地址，具体配置步骤如下。

```
[Device-vsys2] interface gigabitethernet 1/0/3
[Device-vsys2-GigabitEthernet1/0/3] ip address 192.168.2.1 24
[Device-vsys2-GigabitEthernet1/0/3] quit
```

根据组网图中规划的信息，将接口加入对应的安全域，具体配置步骤如下。

```
[Device-vsys2] security-zone name trust
[Device-vsys2-security-zone-Trust] import interface gigabitethernet 1/0/3
[Device-vsys2-security-zone-Trust] quit
[Device-vsys2] security-zone name untrust
[Device-vsys2-security-zone-Untrust] import interface vsys-interface 3
[Device-vsys2-security-zone-Untrust] quit
```

(2) 配置静态路由

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由，本举例假设 vsys2 内用户访问 Internet 的下一跳为缺省 vSystem，实际使用中请以具体组网情况为准，具体配置步骤如下。

```
[Device-vsys2] ip route-static 0.0.0.0 0 public
```

(3) 配置安全策略

配置名称为 trust-untrust 的安全策略，保证 192.168.2.0/24 网段的用户可以访问 Internet，具体配置步骤如下。

```
[Device-vsys2] security-policy ip
[Device-vsys2-security-policy-ip] rule name trust-untrust
[Device-vsys2-security-policy-ip-1-trust-untrust] source-zone trust
[Device-vsys2-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-vsys2-security-policy-ip-1-trust-untrust] source-ip-subnet 192.168.2.0 24
[Device-vsys2-security-policy-ip-1-trust-untrust] action pass
```

从 vsys2 返回缺省 vSystem。

```
[Device-vsys2-security-policy-ip-1-trust-untrust] return
<Device-vsys2> quit
[Device]
```

6. 验证配置

(1) 在 Device 上查看所配 vSystem 是否存在并且运转正常。（此时，Device 上应该有三台处于正常工作 active 状态的 vSystem）

```
[Device] display vsys

ID       Name          Status            Description
```

```
1        Admin         Active            Default
2        vsys1         Active            vsys-1
3        vsys2         Active            vsys-2
```

(2) 位于 192.168.1.128/25 网段的用户可以成功访问 Internet。

```
C:\> ping 3.3.3.3
Pinging 3.3.3.3 with 32 bytes of data:
Reply from 3.3.3.3: bytes=32 time=51ms TTL=255
Reply from 3.3.3.3: bytes=32 time=44ms TTL=255
Reply from 3.3.3.3: bytes=32 time=1ms TTL=255
Reply from 3.3.3.3: bytes=32 time=1ms TTL=255
Ping statistics for 3.3.3.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 51ms, Average = 24ms
```

(3) 位于 192.168.2./24 网段的用户可以成功访问 Internet。

```
C:\> ping 3.3.3.3
Pinging 3.3.3.3 with 32 bytes of data:
Reply from 3.3.3.3: bytes=32 time=25ms TTL=255
Reply from 3.3.3.3: bytes=32 time=36ms TTL=255
Reply from 3.3.3.3: bytes=32 time=1ms TTL=255
Reply from 3.3.3.3: bytes=32 time=1ms TTL=255
Ping statistics for 3.3.3.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 36ms, Average = 16ms
```

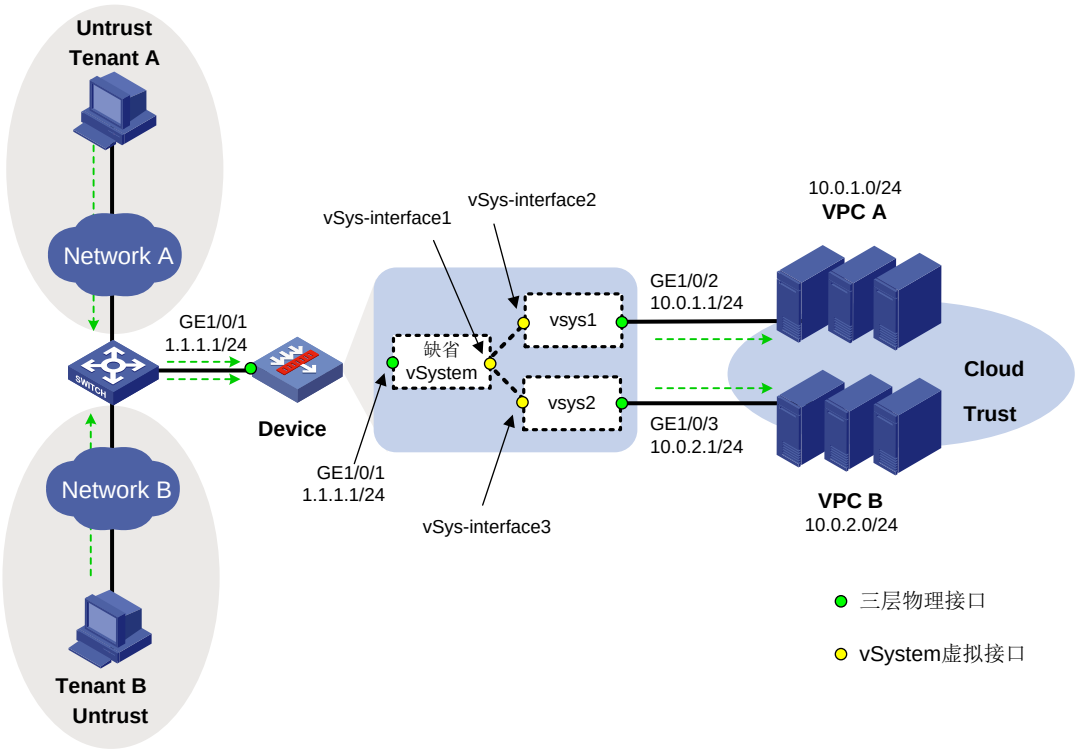
1.11.5 通过 vSystem 实现云计算网关

1. 组网需求

某云计算中心在其网络出口部署了一台 Device 作为安全网关，为其租户提供可定制的安全服务。现有云租户 A 和 B，需要通过独立的公网 IP（1.1.1.2 和 1.1.1.3）访问属于自己的 Web 服务器。云租户 A 和 B 的业务量不同，需要在 Device 上为二者配置不同规模的系统资源。

2. 组网图

图1-6 通过 vSystem 实现云计算网关组网图



3. 配置缺省 vSystem

(1) 创建 vSystem，并为其分配接口

创建 vsys1，设置描述信息。

```
<Device> system-view
[Device] vsys vsys1
[Device-vsys-2-vsys1] description vsys-1
```

将接口 GigabitEthernet1/0/2 分配给 vsys1。

```
[Device-vsys-2-vsys1] allocate interface gigabitethernet 1/0/2
Some configurations on the interface are removed.
[Device-vsys-2-vsys1] quit
```

创建 vsys2，设置描述信息。

```
[Device] vsys vsys2
[Device-vsys-3-vsys2] description vsys-2
```

将接口 GigabitEthernet1/0/3 分配给 vsys2。

```
[Device-vsys-3-vsys2] allocate interface gigabitethernet 1/0/3
Some configurations on the interface are removed.
[Device-vsys-3-vsys2] quit
```

(2) 配置接口 IP 地址及其所属安全域

根据组网图中规划的信息，配置各接口的 IP 地址，具体配置步骤如下。

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 1.1.1.1 24
[Device-GigabitEthernet1/0/1] quit
```

根据组网图中规划的信息，将接口加入对应的安全域，具体配置步骤如下。

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface vsys-interface 1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Device-security-zone-Untrust] quit
```

(3) 配置静态路由

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由，本举例假设内网报文通过缺省 vSystem 到达 Internet 的下一跳 IP 地址为 1.1.1.2，实际使用中请以具体组网情况为准，具体配置步骤如下。

```
[Device] ip route-static 0.0.0.0 0 1.1.1.2
```

配置静态路由，将外网用户访问 VPC A 的流量引入 vSystem vsys1。

```
[Device] ip route-static 10.0.1.0 24 vpn-instance vsys1
```

配置静态路由，将外网用户访问 VPC B 的流量引入 vSystem vsys2。

```
[Device] ip route-static 10.0.2.0 24 vpn-instance vsys2
```

(4) 配置安全策略

配置名称为 untrust-trust 的安全策略，保证外网用户可以访问内网服务器，具体配置步骤如下。

```
[Device] security-policy ip
[Device-security-policy-ip] rule name untrust-trust
[Device-security-policy-ip-1-untrust-trust] source-zone untrust
[Device-security-policy-ip-1-untrust-trust] destination-zone trust
[Device-security-policy-ip-1-untrust-trust] destination-ip-subnet 10.0.1.0 24
[Device-security-policy-ip-1-untrust-trust] destination-ip-subnet 10.0.2.0 24
[Device-security-policy-ip-1-untrust-trust] action pass
[Device-security-policy-ip-1-untrust-trust] quit
[Device-security-policy-ip] quit
```

(5) 配置 NAT 功能

配置 NAT 内部服务器，允许外网用户使用 IP 地址 1.1.1.2、端口号 8080 访问内网 IP 地址为 10.0.1.2 的服务器，允许外网用户使用 IP 地址 1.1.1.3、端口号 8080 访问内网 IP 地址为 10.0.2.2 的服务器。

```
[Device] interface gigabitethernet 1/0/1
```

```
[Device-GigabitEthernet1/0/1] nat server protocol tcp global 1.1.1.2 8080 inside 10.0.1.2 http
[Device-GigabitEthernet1/0/1] nat server protocol tcp global 1.1.1.3 8080 inside 10.0.2.2 http
[Device-GigabitEthernet1/0/1] quit
```

4. 配置 vSystem vsys1

(1) 登录 vsys1，配置接口 IP 地址及其所属安全域

登录 vsys1。

```
[Device] switchto vsys vsys1
<Device-vsys1> system-view
```

根据组网图中规划的信息，配置各接口的 IP 地址，具体配置步骤如下。

```
[Device-vsys1] interface gigabitethernet 1/0/2
[Device-vsys1-GigabitEthernet1/0/2] ip address 10.0.1.1 24
[Device-vsys1-GigabitEthernet1/0/2] quit
```

根据组网图中规划的信息，将接口加入对应的安全域，具体配置步骤如下。

```
[Device-vsys1] security-zone name trust
[Device-vsys1-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-vsys1-security-zone-Trust] quit
[Device-vsys1] security-zone name untrust
[Device-vsys1-security-zone-Untrust] import interface vsys-interface 2
[Device-vsys1-security-zone-Untrust] quit
```

(2) 配置静态路由

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由，本举例假设 vsys1 内用户访问 Internet 的下一跳为缺省 vSystem，实际使用中请以具体组网情况为准，具体配置步骤如下。

```
[Device-vsys1] ip route-static 0.0.0.0 0 public
```

(3) 配置安全策略

配置名称为 untrust-trust 的安全策略，保证外网用户可以访问 VPC A，具体配置步骤如下。

```
[Device-vsys1] security-policy ip
[Device-vsys1-security-policy-ip] rule name untrust-trust
[Device-vsys1-security-policy-ip-1-untrust-trust] source-zone untrust
[Device-vsys1-security-policy-ip-1-untrust-trust] destination-zone trust
[Device-vsys1-security-policy-ip-1-untrust-trust] destination-ip-subnet 10.0.1.0 24
[Device-vsys1-security-policy-ip-1-untrust-trust] action pass
```

从 vsys1 返回缺省 vSystem。

```
[Device-vsys1-security-policy-ip-1-untrust-trust] return
<Device-vsys1> quit
```

```
[Device]
```

5. 配置 vSystem vsys2

(1) 登录 vsys2，配置接口 IP 地址及其所属安全域

登录 vsys2。

```
[Device] switchto vsys vsys2
<Device-vsys2> system-view
```

根据组网图中规划的信息，配置各接口的 IP 地址，具体配置步骤如下。

```
[Device-vsys2] interface gigabitethernet 1/0/3
[Device-vsys2-GigabitEthernet1/0/3] ip address 10.0.2.1 24
[Device-vsys2-GigabitEthernet1/0/3] quit
```

根据组网图中规划的信息，将接口加入对应的安全域，具体配置步骤如下。

```
[Device-vsys2] security-zone name trust
[Device-vsys2-security-zone-Trust] import interface gigabitethernet 1/0/3
[Device-vsys2-security-zone-Trust] quit
[Device-vsys2] security-zone name untrust
[Device-vsys2-security-zone-Untrust] import interface vsys-interface 3
[Device-vsys2-security-zone-Untrust] quit
```

(2) 配置静态路由

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由，本举例假设 vsys2 内用户访问 Internet 的下一跳为缺省 vSystem，实际使用中请以具体组网情况为准，具体配置步骤如下。

```
[Device-vsys2] ip route-static 0.0.0.0 0 public
```

(3) 配置安全策略

配置名称为 untrust-trust 的安全策略，保证外网用户可以访问 VPC B，具体配置步骤如下。

```
[Device-vsys2] security-policy ip
[Device-vsys2-security-policy-ip] rule name untrust-trust
[Device-vsys2-security-policy-ip-1-untrust-trust] source-zone untrust
[Device-vsys2-security-policy-ip-1-untrust-trust] destination-zone trust
[Device-vsys2-security-policy-ip-1-untrust-trust] destination-ip-subnet 10.0.2.0 24
[Device-vsys2-security-policy-ip-1-untrust-trust] action pass
```

从 vsys2 返回缺省 vSystem。

```
[Device-vsys2-security-policy-ip-1-untrust-trust] return
<Device-vsys2> quit
[Device]
```

6. 验证配置

(1) 在 Device 上查看所配 vSystem 是否存在并且运转正常。（此时，Device 上应该有三台处于正常工作 active 状态的 vSystem）

```
[Device] display vsys

ID       Name          Status          Description
1        Admin         Active          Default
2        vsys1         Active          vsys-1
3        vsys2         Active          vsys-2
```

(2) 租户 A 可以通过 URL http://1.1.1.2:8080 访问位于 VPC A 中的服务器 10.0.1.2。

(3) 租户 B 可以通过 URL http://1.1.1.3:8080 访问位于 VPC B 中的服务器 10.0.2.2。

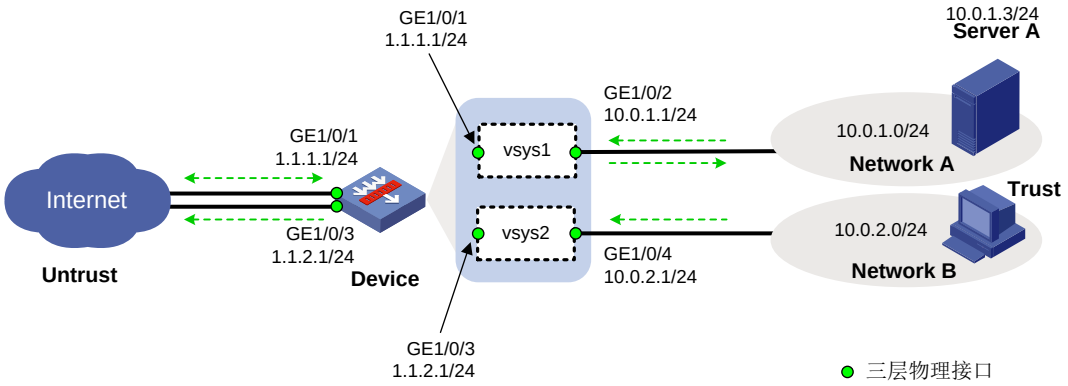
1.11.6 Internet 用户通过非缺省 vSystem 的公网接口访问内部服务器

1. 组网需求

某公司网络出口部署了一台 Device 作为安全网关，其内部划分为网络 A 和网络 B，网络 A 和网络 B 之间相互隔离，其用户不能互访。网络 A 和网络 B 各自通过独立的公网接口接入 Internet。在网络 A 中还部署了一台 Server A，位于 Internet 中的用户可以通过独立的公网 IP（1.1.1.2）对其进行 Web 访问。

2. 组网图

图1-7 Internet 用户通过非缺省 vSystem 的公网接口访问内部服务器组网图



3. 配置 vSystem vsys1

(1) 创建 vSystem vsys1，并为其分配接口

创建 vsys1，设置描述信息。

```
<Device> system-view
[Device] vsys vsys1
[Device-vsys-2-vsys1] description vsys-1
```

将接口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/2 分配给 vsys1。

```
[Device-vsys-2-vsys1] allocate interface gigabitethernet 1/0/1
Some configurations on the interface are removed.
[Device-vsys-2-vsys1] allocate interface gigabitethernet 1/0/2
Some configurations on the interface are removed.
```

```
[Device-vsys-2-vsys1] quit
```

(2) 登录 vsys1，配置接口 IP 地址及其所属安全域

登录 vsys1。

```
[Device] switchto vsys vsys1
<Device-vsys1> system-view
```

根据组网图中规划的信息，配置各接口的 IP 地址，具体配置步骤如下。

```
[Device-vsys1] interface gigabitethernet 1/0/1
[Device-vsys1-GigabitEthernet1/0/1] ip address 1.1.1.1 24
[Device-vsys1-GigabitEthernet1/0/1] quit
[Device-vsys1] interface gigabitethernet 1/0/2
[Device-vsys1-GigabitEthernet1/0/2] ip address 10.0.1.1 24
[Device-vsys1-GigabitEthernet1/0/2] quit
```

根据组网图中规划的信息，将接口加入对应的安全域，具体配置步骤如下。

```
[Device-vsys1] security-zone name trust
[Device-vsys1-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-vsys1-security-zone-Trust] quit
[Device-vsys1] security-zone name untrust
[Device-vsys1-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Device-vsys1-security-zone-Untrust] quit
```

(3) 配置静态路由

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由，本举例假设 vsys1 内用户访问 Internet 的下一跳 IP 地址为 1.1.1.2，实际使用中请以具体组网情况为准，具体配置步骤如下。

```
[Device-vsys1] ip route-static 0.0.0.0 0 1.1.1.2
```

(4) 配置安全策略

配置名称为 trust-untrust 的安全策略，保证内网用户可以访问 Internet，具体配置步骤如下。

```
[Device-vsys1] security-policy ip
[Device-vsys1-security-policy-ip] rule name trust-untrust
[Device-vsys1-security-policy-ip-1-trust-untrust] source-zone trust
[Device-vsys1-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-vsys1-security-policy-ip-1-trust-untrust] source-ip-subnet 10.0.1.0 24
[Device-vsys1-security-policy-ip-1-trust-untrust] action pass
[Device-vsys1-security-policy-ip-1-trust-untrust] quit
```

配置名称为 untrust-trust 的安全策略，保证外网用户可以访问 Server A，具体配置步骤如下。

```
[Device-vsys1-security-policy-ip] rule name untrust-trust
[Device-vsys1-security-policy-ip-2-untrust-trust] source-zone untrust
```

```
[Device-vsys1-security-policy-ip-2-untrust-trust] destination-zone trust
[Device-vsys1-security-policy-ip-2-untrust-trust] destination-ip-host 10.0.1.3
[Device-vsys1-security-policy-ip-2-untrust-trust] action pass
[Device-vsys1-security-policy-ip-2-untrust-trust] quit
[Device-vsys1-security-policy-ip] quit
```

(5) 配置 NAT 功能

创建地址组 1，包含公网地址 1.1.1.100。

```
[Device-vsys1] nat address-group 1
[Device-vsys1-address-group-1] address 1.1.1.100 1.1.1.100
[Device-vsys1-address-group-1] quit
```

配置 ACL 2000，仅允许对来自网络 A 的报文进行地址转换。

```
[Device-vsys1] acl basic 2000
[Device-vsys1-acl-ipv4-basic-2000] rule permit source 10.0.1.0 0.0.0.255
[Device-vsys1-acl-ipv4-basic-2000] quit
```

配置 NAT 出方向地址转换，允许网络 A 内的用户使用地址组 1 中的地址访问 Internet。

```
[Device-vsys1] interface gigabitethernet 1/0/1
[Device-vsys1-GigabitEthernet1/0/1] nat outbound 2000 address-group 1
```

配置 NAT 内部服务器，允许外网用户使用 IP 地址 1.1.1.2、端口号 8080 访问 Server A。

```
[Device-vsys1-GigabitEthernet1/0/1] nat server protocol tcp global 1.1.1.2 8080 inside
10.0.1.3 http
```

从 vsys1 返回缺省 vSystem。

```
[Device-vsys1-GigabitEthernet1/0/1] return
<Device-vsys1> quit
[Device]
```

4. 配置 vSystem vsys2

(1) 创建 vSystem vsys2，并为其分配接口

创建 vsys2，设置描述信息。

```
<Device> system-view
[Device] vsys vsys2
[Device-vsys-3-vsys2] description vsys-2
```

将接口 GigabitEthernet1/0/3 和 GigabitEthernet1/0/4 分配给 vsys2。

```
[Device-vsys-3-vsys2] allocate interface gigabitethernet 1/0/3
Some configurations on the interface are removed.
[Device-vsys-3-vsys2] allocate interface gigabitethernet 1/0/4
Some configurations on the interface are removed.
[Device-vsys-3-vsys2] quit
```

(2) 登录 vsys2，配置接口 IP 地址及其所属安全域

登录 vsys2。

```
[Device] switchto vsys vsys2
<Device-vsys2> system-view
```

根据组网图中规划的信息，配置各接口的 IP 地址，具体配置步骤如下。

```
[Device-vsys2] interface gigabitethernet 1/0/3
[Device-vsys2-GigabitEthernet1/0/3] ip address 1.1.2.1 24
[Device-vsys2-GigabitEthernet1/0/3] quit
[Device-vsys2] interface gigabitethernet 1/0/4
[Device-vsys2-GigabitEthernet1/0/4] ip address 10.0.2.1 24
[Device-vsys2-GigabitEthernet1/0/4] quit
```

根据组网图中规划的信息，将接口加入对应的安全域，具体配置步骤如下。

```
[Device-vsys2] security-zone name trust
[Device-vsys2-security-zone-Trust] import interface gigabitethernet 1/0/4
[Device-vsys2-security-zone-Trust] quit
[Device-vsys2] security-zone name untrust
[Device-vsys2-security-zone-Untrust] import interface gigabitethernet 1/0/3
[Device-vsys2-security-zone-Untrust] quit
```

(3) 配置静态路由

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由，本举例假设 vsys2 内用户访问 Internet 的下一跳 IP 地址为 1.1.2.2，实际使用中请以具体组网情况为准，具体配置步骤如下。

```
[Device-vsys2] ip route-static 0.0.0.0 0 1.1.2.2
```

(4) 配置安全策略

配置名称为 trust-untrust 的安全策略，保证内网用户可以访问 Internet，具体配置步骤如下。

```
[Device-vsys2] security-policy ip
[Device-vsys2-security-policy-ip] rule name trust-untrust
[Device-vsys2-security-policy-ip-1-trust-untrust] source-zone trust
[Device-vsys2-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-vsys2-security-policy-ip-1-trust-untrust] source-ip-subnet 10.0.2.0 24
[Device-vsys2-security-policy-ip-1-trust-untrust] action pass
[Device-vsys2-security-policy-ip-1-trust-untrust] quit
```

(5) 配置 NAT 功能

创建地址组 1，包含公网地址 1.1.1.101。

```
[Device-vsys2] nat address-group 1
[Device-vsys2-address-group-1] address 1.1.1.100 1.1.1.101
[Device-vsys2-address-group-1] quit
```

配置 ACL 2000，仅允许对来自网络 B 的报文进行地址转换。

```
[Device-vsys2] acl basic 2000
[Device-vsys2-acl-ipv4-basic-2000] rule permit source 10.0.2.0 0.0.0.255
[Device-vsys2-acl-ipv4-basic-2000] quit
```

配置 NAT 出方向地址转换，允许网络 B 内的用户使用地址组 1 中的地址访问 Internet。

```
[Device-vsys2] interface gigabitethernet 1/0/3
[Device-vsys2-GigabitEthernet1/0/3] nat outbound 2000 address-group 1
```

从 vsys2 返回缺省 vSystem。

```
[Device-vsys2-GigabitEthernet1/0/3] return
<Device-vsys2> quit
[Device]
```

5. 验证配置

(1) 在 Device 上查看所配 vSystem 是否存在并且运转正常。（此时，Device 上应该有三台处于正常工作 active 状态的 vSystem）

```
[Device] display vsys

ID       Name          Status            Description
1        Admin         Active            Default
2        vsys1         Active            vsys-1
3        vsys2         Active            vsys-2
```

(2) Internet 中的用户可以通过 URL http://1.1.1.2:8080 访问 Server A。

(3) 网络 A 内的用户可以成功访问 Internet。

```
C:\> ping 3.3.3.3
Pinging 3.3.3.3 with 32 bytes of data:
Reply from 3.3.3.3: bytes=32 time=51ms TTL=255
Reply from 3.3.3.3: bytes=32 time=44ms TTL=255
Reply from 3.3.3.3: bytes=32 time=1ms TTL=255
Reply from 3.3.3.3: bytes=32 time=1ms TTL=255
Ping statistics for 3.3.3.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 51ms, Average = 24ms
```

(4) 网络 B 内的用户可以成功访问 Internet。

```
C:\> ping 3.3.3.3
Pinging 3.3.3.3 with 32 bytes of data:
Reply from 3.3.3.3: bytes=32 time=25ms TTL=255
Reply from 3.3.3.3: bytes=32 time=36ms TTL=255
Reply from 3.3.3.3: bytes=32 time=1ms TTL=255
Reply from 3.3.3.3: bytes=32 time=1ms TTL=255
```

```
Ping statistics for 3.3.3.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 36ms, Average = 16ms
```

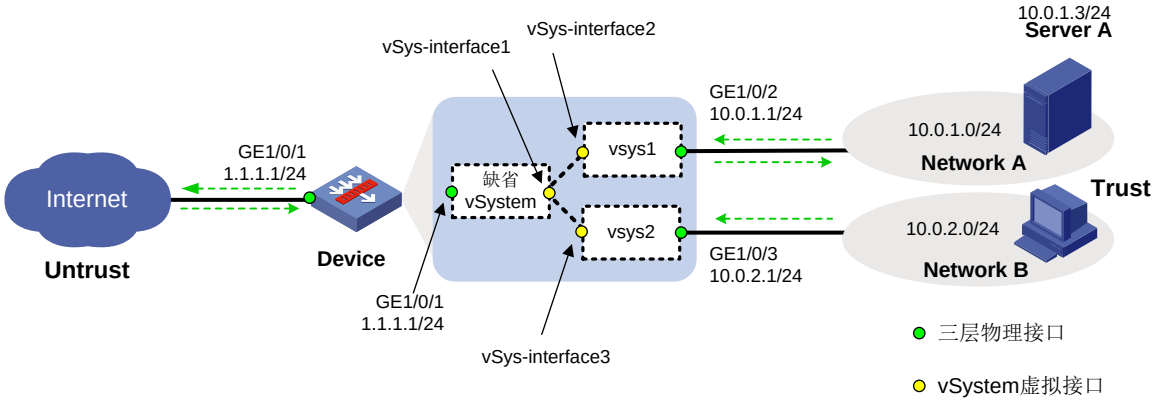
1.11.7 Internet 用户通过缺省 vSystem 的公网接口访问内部服务器

1. 组网需求

某公司网络出口部署了一台 Device 作为安全网关，其内部划分为网络 A 和网络 B，网络 A 和网络 B 之间相互隔离，其用户不能互访。网络 A 和网络 B 通过 Device 上的一个公网接口接入 Internet，其中网络 A 内的用户使用公网地址 1.1.1.100，网络 B 内的用户使用公网地址 1.1.1.101。在网络 A 中还部署了一台 Server A，位于 Internet 中的用户可以通过独立的公网 IP（1.1.1.2）对其进行 Web 访问。

2. 组网图

图1-8 Internet 用户通过缺省 vSystem 的公网接口访问内部服务器组网图



3. 配置缺省 vSystem

(1) 创建 vSystem，并为其分配接口

创建 vsys1，设置描述信息。

```
<Device> system-view
[Device] vsys vsys1
[Device-vsys-2-vsys1] description vsys-1
```

将接口 GigabitEthernet1/0/2 分配给 vsys1。

```
[Device-vsys-2-vsys1] allocate interface gigabitethernet 1/0/2
Some configurations on the interface are removed.
[Device-vsys-2-vsys1] quit
```

创建 vsys2，设置描述信息。

```
[Device] vsys vsys2
[Device-vsys-3-vsys2] description vsys-2
```

将接口 GigabitEthernet1/0/3 分配给 vsys2。

```
[Device-vsys-3-vsys2] allocate interface gigabitethernet 1/0/3
Some configurations on the interface are removed.
[Device-vsys-3-vsys2] quit
```

(2) 配置接口 IP 地址及其所属安全域

根据组网图中规划的信息，配置各接口的 IP 地址，具体配置步骤如下。

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] ip address 1.1.1.1 24
[Device-GigabitEthernet1/0/1] quit
```

根据组网图中规划的信息，将接口加入对应的安全域，具体配置步骤如下。

```
[Device] security-zone name trust
[Device-security-zone-Trust] import interface vsys-interface 1
[Device-security-zone-Trust] quit
[Device] security-zone name untrust
[Device-security-zone-Untrust] import interface gigabitethernet 1/0/1
[Device-security-zone-Untrust] quit
```

(3) 配置静态路由

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由，本举例假设内网报文通过缺省 vSystem 到达 Internet 的下一跳 IP 地址为 1.1.1.2，实际使用中请以具体组网情况为准，具体配置步骤如下。

```
[Device] ip route-static 0.0.0.0 0 1.1.1.2
```

配置静态路由，将外网用户访问 VPC A 的流量引入 vSystem vsys1。

```
[Device] ip route-static 10.0.1.0 24 vpn-instance vsys1
```

配置静态路由，将外网用户访问 VPC B 的流量引入 vSystem vsys2。

```
[Device] ip route-static 10.0.2.0 24 vpn-instance vsys2
```

(4) 配置安全策略

配置名称为 untrust-trust 的安全策略，保证外网用户可以访问 Server A，具体配置步骤如下。

```
[Device] security-policy ip
[Device-security-policy-ip] rule name untrust-trust
[Device-security-policy-ip-1-untrust-trust] source-zone untrust
[Device-security-policy-ip-1-untrust-trust] destination-zone trust
[Device-security-policy-ip-1-untrust-trust] destination-ip-host 10.0.1.3
[Device-security-policy-ip-1-untrust-trust] action pass
[Device-security-policy-ip-1-untrust-trust] quit
```

配置名称为 trust-untrust 的安全策略，保证内网用户可以访问 Internet，具体配置步骤如下。

```
[Device-security-policy-ip] rule name trust-untrust
[Device-security-policy-ip-2-trust-untrust] source-zone trust
```

```
[Device-security-policy-ip-2-trust-untrust] destination-zone untrust
[Device-security-policy-ip-2-trust-untrust] source-ip-subnet 10.0.1.0 24
[Device-security-policy-ip-2-trust-untrust] source-ip-subnet 10.0.2.0 24
[Device-security-policy-ip-2-trust-untrust] action pass
[Device-security-policy-ip-2-trust-untrust] quit
[Device-security-policy-ip] quit
```

(5) 配置 NAT 功能

创建地址组 1，包含公网地址 1.1.1.100；创建地址组 2，包含公网地址 1.1.1.101。

```
[Device] nat address-group 1
[Device-address-group-1] address 1.1.1.100 1.1.1.100
[Device-address-group-1] quit
[Device] nat address-group 2
[Device-address-group-2] address 1.1.1.101 1.1.1.101
[Device-address-group-2] quit
```

配置 ACL 2000，仅允许对来自网络 A 的报文进行地址转换；配置 ACL 2001，仅允许对来自网络 B 的报文进行地址转换。

```
[Device] acl basic 2000
[Device-acl-ipv4-basic-2000] rule permit source 10.0.1.0 0.0.0.255
[Device-acl-ipv4-basic-2000] quit
[Device] acl basic 2001
[Device-acl-ipv4-basic-2001] rule permit source 10.0.2.0 0.0.0.255
[Device-acl-ipv4-basic-2001] quit
```

配置 NAT 出方向地址转换，允许网络 A 内的用户使用地址组 1 中的地址访问 Internet，允许网络 B 内的用户使用地址组 2 中的地址访问 Internet。

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] nat outbound 2000 address-group 1
[Device-GigabitEthernet1/0/1] nat outbound 2001 address-group 2
```

配置 NAT 内部服务器，允许外网用户使用 IP 地址 1.1.1.2、端口号 8080 访问 Server A。

```
[Device-GigabitEthernet1/0/1] nat server protocol tcp global 1.1.1.2 8080 inside
10.0.1.3 http
[Device-GigabitEthernet1/0/1] quit
```

4. 配置 vSystem vsys1

(1) 登录 vsys1，配置接口 IP 地址及其所属安全域

登录 vsys1。

```
[Device] switchto vsys vsys1
<Device-vsys1> system-view
```

根据组网图中规划的信息，配置各接口的 IP 地址，具体配置步骤如下。

```
[Device-vsys1] interface gigabitethernet 1/0/2
```

```
[Device-vsys1-GigabitEthernet1/0/2] ip address 10.0.1.1 24
[Device-vsys1-GigabitEthernet1/0/2] quit
```

根据组网图中规划的信息，将接口加入对应的安全域，具体配置步骤如下。

```
[Device-vsys1] security-zone name trust
[Device-vsys1-security-zone-Trust] import interface gigabitethernet 1/0/2
[Device-vsys1-security-zone-Trust] quit
[Device-vsys1] security-zone name untrust
[Device-vsys1-security-zone-Untrust] import interface vsys-interface 2
[Device-vsys1-security-zone-Untrust] quit
```

(2) 配置静态路由

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由，本举例假设 vsys1 内用户访问 Internet 的下一跳为缺省 vSystem，实际使用中请以具体组网情况为准，具体配置步骤如下。

```
[Device-vsys1] ip route-static 0.0.0.0 0 public
```

(3) 配置安全策略

配置名称为 untrust-trust 的安全策略，保证外网用户可以访问 Server A，具体配置步骤如下。

```
[Device-vsys1] security-policy ip
[Device-vsys1-security-policy-ip] rule name untrust-trust
[Device-vsys1-security-policy-ip-1-untrust-trust] source-zone untrust
[Device-vsys1-security-policy-ip-1-untrust-trust] destination-zone trust
[Device-vsys1-security-policy-ip-1-untrust-trust] destination-ip-host 10.0.1.3
[Device-vsys1-security-policy-ip-1-untrust-trust] action pass
[Device-vsys1-security-policy-ip-1-untrust-trust] quit
```

配置名称为 trust-untrust 的安全策略，保证内网用户可以访问 Internet，具体配置步骤如下。

```
[Device-vsys1-security-policy-ip] rule name trust-untrust
[Device-vsys1-security-policy-ip-2-trust-untrust] source-zone trust
[Device-vsys1-security-policy-ip-2-trust-untrust] destination-zone untrust
[Device-vsys1-security-policy-ip-2-trust-untrust] source-ip-subnet 10.0.1.0 24
[Device-vsys1-security-policy-ip-2-trust-untrust] action pass
```

从 vsys1 返回缺省 vSystem。

```
[Device-vsys1-security-policy-ip-2-trust-untrust] return
<Device-vsys1> quit
[Device]
```

5. 配置 vSystem vsys2

(1) 登录 vsys2，配置接口 IP 地址及其所属安全域

登录 vsys2。

```
[Device] switchto vsys vsys2
<Device-vsys2> system-view
```

根据组网图中规划的信息，配置各接口的 IP 地址，具体配置步骤如下。

```
[Device-vsys2] interface gigabitethernet 1/0/3
[Device-vsys2-GigabitEthernet1/0/3] ip address 10.0.2.1 24
[Device-vsys2-GigabitEthernet1/0/3] quit
```

根据组网图中规划的信息，将接口加入对应的安全域，具体配置步骤如下。

```
[Device-vsys2] security-zone name trust
[Device-vsys2-security-zone-Trust] import interface gigabitethernet 1/0/3
[Device-vsys2-security-zone-Trust] quit
[Device-vsys2] security-zone name untrust
[Device-vsys2-security-zone-Untrust] import interface vsys-interface 3
[Device-vsys2-security-zone-Untrust] quit
```

(2) 配置静态路由

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由，本举例假设 vsys2 内用户访问 Internet 的下一跳为缺省 vSystem，实际使用中请以具体组网情况为准，具体配置步骤如下。

```
[Device-vsys2] ip route-static 0.0.0.0 0 public
```

(3) 配置安全策略

配置名称为 trust-untrust 的安全策略，保证内网用户可以访问 Internet，具体配置步骤如下。

```
[Device-vsys2] security-policy ip
[Device-vsys2-security-policy-ip] rule name trust-untrust
[Device-vsys2-security-policy-ip-1-trust-untrust] source-zone trust
[Device-vsys2-security-policy-ip-1-trust-untrust] destination-zone untrust
[Device-vsys2-security-policy-ip-1-trust-untrust] source-ip-subnet 10.0.2.0 24
[Device-vsys2-security-policy-ip-1-trust-untrust] action pass
```

从 vsys2 返回缺省 vSystem。

```
[Device-vsys2-security-policy-ip-1-trust-untrust] return
<Device-vsys2> quit
[Device]
```

6. 验证配置

(1) 在 Device 上查看所配 vSystem 是否存在并且运转正常。（此时，Device 上应该有三台处于正常工作 active 状态的 vSystem）

```
[Device] display vsys
```

```
ID       Name          Status            Description
1        Admin         Active            Default
2        vsys1         Active            vsys-1
3        vsys2         Active            vsys-2
```

(2) Internet 中的用户可以通过 URL http://1.1.1.2:8080 访问 Server A。

(3) 网络 A 内的用户可以成功访问 Internet。

```
C:\> ping 3.3.3.3
Pinging 3.3.3.3 with 32 bytes of data:
Reply from 3.3.3.3: bytes=32 time=51ms TTL=255
Reply from 3.3.3.3: bytes=32 time=44ms TTL=255
Reply from 3.3.3.3: bytes=32 time=1ms TTL=255
Reply from 3.3.3.3: bytes=32 time=1ms TTL=255
Ping statistics for 3.3.3.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 51ms, Average = 24ms
```

(4) 网络 B 内的用户可以成功访问 Internet。

```
C:\> ping 3.3.3.3
Pinging 3.3.3.3 with 32 bytes of data:
Reply from 3.3.3.3: bytes=32 time=25ms TTL=255
Reply from 3.3.3.3: bytes=32 time=36ms TTL=255
Reply from 3.3.3.3: bytes=32 time=1ms TTL=255
Reply from 3.3.3.3: bytes=32 time=1ms TTL=255
Ping statistics for 3.3.3.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 36ms, Average = 16ms
```

目 录

1 Context ····· 1-1
1.1 Context 简介 ····· 1-1
1.1.1 Context 的应用 ····· 1-1
1.1.2 缺省 Context 和非缺省 Context ····· 1-1
1.2 Context 配置限制和指导 ····· 1-2
1.3 Context 配置任务简介 ····· 1-2
1.4 创建 Context ····· 1-3
1.5 将 Context 进驻安全引擎 ····· 1-3
1.5.1 功能简介 ····· 1-3
1.5.2 配置安全引擎组 ····· 1-3
1.5.3 将 Context 进驻安全引擎组 ····· 1-4
1.6 为 Context 分配资源 ····· 1-5
1.6.1 为 Context 分配接口 ····· 1-5
1.6.2 为 Context 分配 VLAN ····· 1-6
1.6.3 为 Context 分配 VXLAN ····· 1-6
1.7 限制 Context 的资源使用 ····· 1-7
1.7.1 限制 Context 出方向的吞吐量 ····· 1-7
1.7.2 限制 Context 对象策略规则总数 ····· 1-8
1.7.3 限制 Context 安全策略规则总数 ····· 1-9
1.7.4 限制 Context 会话并发数 ····· 1-9
1.7.5 限制 Context 会话新建速率 ····· 1-10
1.7.6 限制 Context 的 SSL VPN 登录用户数 ····· 1-10
1.8 启动 Context ····· 1-11
1.9 为 Context 分配 CPU/磁盘/内存资源 ····· 1-11
1.9.1 功能简介 ····· 1-11
1.9.2 为 Context 分配 CPU 权重 ····· 1-11
1.9.3 为 Context 分配磁盘空间上限 ····· 1-12
1.9.4 为 Context 分配内存空间上限 ····· 1-12
1.10 访问和管理 Context ····· 1-13
1.11 配置 Context 限速功能 ····· 1-13
1.11.1 配置 Context 限制广播报文速率 ····· 1-13
1.11.2 配置 Context 限制组播报文速率 ····· 1-14
1.11.3 开启 Context 限速丢包日志功能 ····· 1-15

1.12 配置 CPU 核的攻击防范阈值 ····· 1-15
1.13 收集各 Context 的日志信息 ····· 1-16
1.14 配置 Context 支持跨 VPC 流量互通 ····· 1-16
1.15 Context 显示和维护 ····· 1-17
1.16 Context 典型配置举例 ····· 1-19
1.16.1 Context 基本组网配置举例（独立运行模式） ····· 1-19
1.16.2 Context 支持跨 VPC 流量互通典型配置举例（独立运行模式） ····· 1-24
1.16.3 通过 Context 实现云计算中心网关配置举例（独立运行模式） ····· 1-27

1 Context

1.1 Context简介

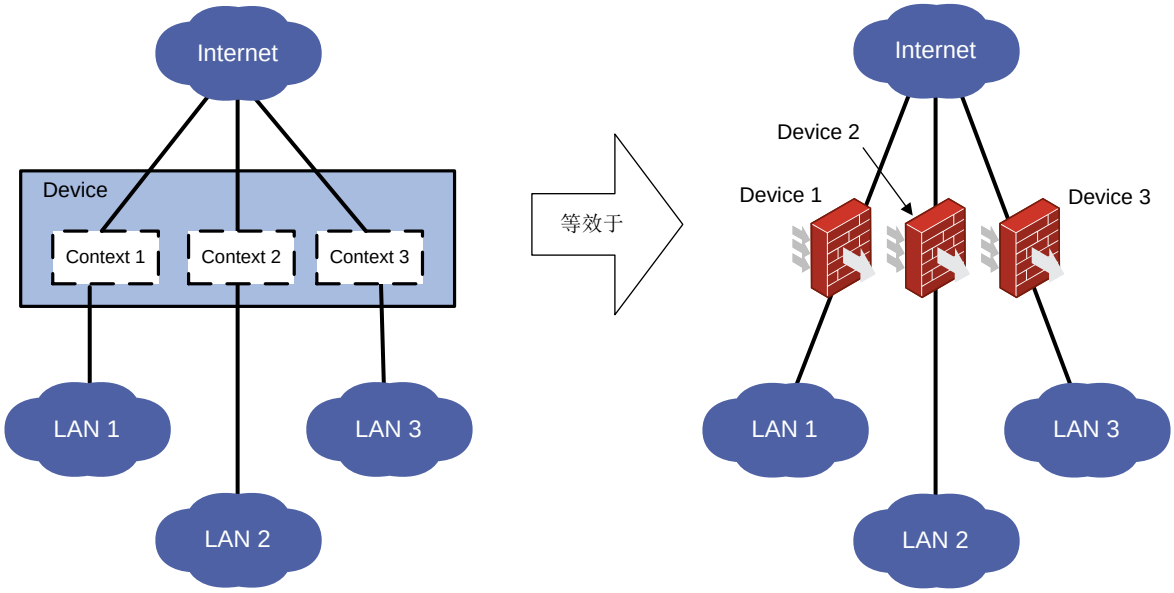
通过虚拟化技术将一台物理设备划分成多台逻辑设备，每台逻辑设备就称为一个 Context。每个 Context 拥有自己专属的软硬件资源，独立运行。

对于用户来说，每个 Context 就是一台独立的设备，方便管理和维护；对于管理者来说，可以将一台物理设备虚拟成多台逻辑设备供不同的分支机构使用，可以保护现有投资，提高组网灵活性。

1.1.1 Context 的应用

如图 1-1 所示，LAN 1、LAN 2 和 LAN 3 是三个不同的局域网，它们通过同一台设备 Device 连接到外网。通过虚拟化技术，能让一台设备当三台设备使用。具体做法是，在 Device 上创建三个 Context（Context 1、Context 2、Context 3），分别负责 LAN 1、LAN 2、LAN 3 的安全接入。LAN 1、LAN 2、LAN 3 的网络管理员可以（也只能）分别登录到自己的设备进行配置、保存、重启等操作，不会影响其它网络的使用，其效果等同于 LAN 1、LAN 2 和 LAN 3 分别通过各自的设备 Device 1、Device 2、Device 3 接入 Internet。

图1-1 Context 组网示意图



1.1.2 缺省 Context 和非缺省 Context

- 设备支持 Context 功能后，整台物理设备就是一个 Context，称为缺省 Context，如图 1-1 中的 Device。当用户登录物理设备时，实际登录的就是缺省 Context。用户在物理设备上的配置实质就是对缺省 Context 的配置。缺省 Context 的名称为 Admin，编号为 1。缺省 Context 不需要创建，不能删除。

- 与缺省 Context 相对应的是非缺省 Context，如图 1-1 中的 Context 1、Context 2、Context 3。非缺省 Context 是管理员在设备上通过命令行创建的，可分配给不同的接入网络使用。
- 缺省 Context 拥有对整台物理设备的所有权限，它可以使用和管理设备所有的资源。缺省 Context 下可以创建/删除非缺省 Context，给非缺省 Context 分配 CPU 资源/磁盘/内存空间、接口、VLAN、VXLAN，没有分配的 CPU 资源/磁盘/内存空间、接口、VLAN、VXLAN 由缺省 Context 使用和管理。
- 非缺省 Context 下不可再创建/删除非缺省 Context，它只能使用缺省 Context 分配给自己的资源，并在缺省 Context 指定的资源限制范围内工作，不能抢占其他 Context 或者系统剩余的资源。

1.2 Context配置限制和指导

非缺省 Context 中的 DPI 业务功能使用缺省 Context 中的应用层检测引擎对报文进行匹配，当创建、删除、关闭和重启非缺省 Context 时，缺省 Context 中的应用层检测引擎会重新激活，激活期间设备上的所有 Context 均不能对报文进行 DPI 业务处理。

1.3 Context配置任务简介

Context 配置任务如下：

(1) 创建 Context

(2) 将 Context 进驻安全引擎

(3) （可选）为 Context 分配资源
   - 为 Context 分配接口
   - 为 Context 分配 VLAN
   - 为 Context 分配 VXLAN

(4) （可选）限制 Context 的资源使用
   - 限制 Context 出方向的吞吐量
   - 限制 Context 对象策略规则总数
   - 限制 Context 安全策略规则总数
   - 限制 Context 会话并发数
   - 限制 Context 会话新建速率
   - 限制 Context 的 SSL VPN 登录用户数

(5) 启动 Context

(6) （可选）为 Context 分配 CPU/磁盘/内存资源
   - 为 Context 分配 CPU 权重
   - 为 Context 分配磁盘空间上限
   - 为 Context 分配内存空间上限

(7) 访问和管理 Context

(8) （可选）配置 Context 限速功能

(9) （可选）配置 CPU 核的攻击防范阈值

(10) （可选）收集各 Context 的日志信息

(11) （可选）配置 Context 支持跨 VPC 流量互通

1.4 创建Context

1. 配置限制和指导

创建 Context 相当于构造了一台新的设备。

创建 Context 时，通过 **vlan-unshared** 参数可选择是否和其它 Context 共享 VLAN：

- 如果选择和其它 Context 共享 VLAN，需要在缺省 Context 内创建并配置 VLAN，再分配给非缺省 Context。共享 VLAN 由多个 Context 共同所有。VLAN 1 为系统缺省 VLAN，由缺省 Context 独有，不能分配给非缺省 Context。此种方式的非缺省 Context 不支持将以太网接口切换为二层模式，也不支持将二层以太网接口独占方式分配给此种非缺省 Context。
- 如果选择不和其它 Context 共享 VLAN，请登录该 Context，并使用 **vlan** 命令创建 VLAN 2～VLAN 4094。VLAN 1 为缺省 VLAN，用户不能手工创建和删除。Context 各自使用和管理 VLAN，互不干扰。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 创建 Context，并进入 Context 视图。

context *context-name* [**id** *context-id*] [**vlan-unshared**]

缺省情况下，设备上存在缺省 Context，名称为 Admin，编号为 1。

(3) （可选）配置 Context 的描述信息。

description *text*

缺省情况下，缺省 Context 描述信息为 DefaultContext。非缺省 Context 没有配置描述信息

1.5 将Context进驻安全引擎

1.5.1 功能简介

安全引擎是设备上处理安全业务的专有硬件单元，一个安全引擎对应安全业务板上的一个 CPU。如果一个安全业务板只有一个 CPU，则此安全业务板就是一个安全引擎；如果一个安全业务板有多个 CPU，则此安全业务板上也就存在多个安全引擎。Context 创建后必须进驻安全引擎（通过将 Context 进驻安全引擎组来实现），才有实际运行的环境，才能运行业务。

1.5.2 配置安全引擎组

1. 功能简介

安全引擎组是一个逻辑概念，用于组织和管理安全引擎，一个安全引擎组中可添加多个安全引擎。新创建的安全引擎组内没有安全引擎，必须先将安全引擎加入安全引擎组，才能使 Context 进驻到安全引擎组中的所有安全引擎。

在数据层面安全引擎组中的所有安全引擎都会同时处理安全业务和转发报文。在管理层面系统会自动选举一个引擎为主安全引擎，其它均为从安全引擎。从安全引擎以备份身份运行，当主安全引擎不能正常工作时，会将一个从安全引擎升级为新的主安全引擎，替代原主安全引擎工作。

2. 配置限制和指导

注意

缺省安全引擎组中必须至少存在一个安全引擎，否则设备不能正常处理业务。

一个安全引擎只能属于一个安全引擎组。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 创建安全引擎组并进入该安全引擎组视图。

blade-controller-team *blade-controller-team-name* [**id** *blade-controller-team-id*]

缺省情况下，设备上有一个缺省安全引擎组，名称为 Default，编号为 1。

(3) 将安全引擎加入安全引擎组。

（独立运行模式）

location blade-controller slot *slot-number* **cpu** *cpu-number*

（IRF 模式）

location blade-controller chassis *chassis-number* **slot** *slot-number* **cpu** *cpu-number*

缺省情况下，安全引擎插入时会自动加入缺省安全引擎组。

1.5.3 将 Context 进驻安全引擎组

1. 功能简介

为了在 Context 上启动业务，用户必须将 Context 进驻安全引擎组。Context 进驻安全引擎组后，会进驻安全引擎组内的所有安全引擎。

2. 配置限制和指导

Context 和安全引擎组的关系如下：

- 一个 Context 只能进驻一个安全引擎组。如果该 Context 已经进驻一个安全引擎组，请先执行 **undo location blade-controller-team** 命令退出已进驻的安全引擎组，再配置 **location blade-controller-team** 命令，进驻其它安全引擎组。
- 在不同的 Context 视图下执行该命令可以使多个 Context 进驻同一个安全引擎组，安全引擎组和 Context 是一对多的关系。
- 安全引擎组中加入新的安全引擎后，安全引擎组上已进驻的 Context 会自动进驻到新加入的安全引擎上，不需要再次配置。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 Context 视图。

context *context-name*

(3) 将 Context 进驻安全引擎组。

location blade-controller-team *team-id*

缺省情况下，缺省 Context 进驻了所有安全引擎组，非缺省 Context 没有进驻任何安全引擎组。

1.6 为Context分配资源

1.6.1 为 Context 分配接口

1. 接口分配简介

设备上的所有接口都属于缺省 Context，不属于任何非缺省 Context。请给非缺省 Context 分配接口，它才能和网络中的其它设备通信。

为了提高设备接口的利用率，在给 Context 分配接口时，可以选择：

- 独占方式分配（不带 **share** 参数）。使用该方式分配的接口仅归该 Context 所有、使用。用户登录该 Context 后，能查看到该接口，并执行接口支持的所有命令。
- 共享方式分配（带 **share** 参数）：表示将一个接口分配给多个 Context 使用，这些 Context 共享这个物理接口，但是在各个 Context 内会创建一个同名的虚接口，这些虚接口具有不同的 MAC 地址和 IP 地址。设备从共享的物理接口接收报文后交给对应的虚拟接口处理；出方向，虚拟接口处理完报文后，会交给共享的物理接口发送。使用该方式，可以提高设备接口的利用率。通过共享方式分配的接口：
  - 在缺省 Context 内仍然存在该接口，该接口可执行接口支持的所有命令；
  - 在非缺省 Context 内，会新建一个同名接口，用户登录这些 Context 后，能查看到该接口，但只能执行 **description** 以及网络/安全相关的命令。

2. 配置限制和指导

当设备运行在 IRF 模式时，禁止将 IRF 物理端口分配给 Context。

聚合接口的成员接口不能分配给 Context。

冗余口的成员接口不能分配给 Context，当冗余口的成员接口为子接口时，其子接口的主接口也不能分配给 Context。

逻辑接口（如子接口、聚合接口等）仅支持共享方式分配，物理接口支持独占和共享两种方式分配。

如果子接口已经被分配，则不能再分配其父接口；如果父接口已经被分配，则不能再分配其子接口。

如果接口已经被共享分配，则不能再独占分配。需将共享分配配置取消后，才能独占分配。

为使非缺省 Context 之间可以互通，必须在缺省 Context 中将物理接口或逻辑接口以共享方式分配给非缺省 Context。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 Context 视图。

context *context-name*

(3) 为 Context 分配接口。

- 非连续接口配置。

  allocate interface { *interface-type interface-number* }&<1-24> [**share**]

- 连续接口配置。

  allocate interface *interface-type interface-number1* **to** *interface-type interface-number2* [**share**]

缺省情况下，设备上的所有接口都属于缺省 Context，不属于任何非缺省 Context。

1.6.2 为 Context 分配 VLAN

1. 配置限制和指导

创建 Context 时，如果不选择 **vlan-unshared** 参数，则表示和其它 Context 共享 VLAN。

对于共享 VLAN，请先在缺省 Context 内创建 VLAN，再通过 **allocate vlan** 命令将指定 VLAN 分配给指定的 Context 使用。

VLAN 1 不能被共享。

端口的缺省 VLAN 不能被共享。

已经创建了 VLAN 接口的 VLAN 不能被共享。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 Context 视图。

context *context-name*

(3) 为 Context 分配 VLAN。

- 非连续 VLAN 配置。

  allocate vlan *vlan-id*&<1-24>

- 连续 VLAN 配置。

  allocate vlan *vlan-id1* **to** *vlan-id2*

缺省情况下，没有为 Context 分配 VLAN。

1.6.3 为 Context 分配 VXLAN

1. 配置限制和指导

为 Context 分配的 VXLAN 仅归该 Context 所有，其他 Context 不能对其进行使用、配置。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 Context 视图。

context *context-name*

(3) 为 Context 分配 VXLAN。

- 非连续 VXLAN 配置。

  allocate vxlan *vxlan-id*&<1-24>

- 连续 VXLAN 配置。

  allocate vxlan *vxlan-id1* **to** *vxlan-id2*

缺省情况下，没有为 Context 分配 VXLAN。

1.7 限制Context的资源使用

1.7.1 限制 Context 出方向的吞吐量

1. 功能简介

为了防止一个 Context 发送的报文过多而导致其它 Context 发送的报文被丢弃，需要限制 Context 出方向的吞吐量。

一个 Context 出方向的吞吐量，是在其进驻的每个安全引擎内独立计算的，即 Context 在每个进驻的安全引擎上享有相同的吞吐量，多个报文分散在不同引擎处理时，实际吞吐量会大于限制的值。

除此之外，还可以针对 Context 的出方向吞吐量限制开启吞吐量告警功能和吞吐量限速丢包日志功能。

- 出方向吞吐量告警功能：开启此功能并设置告警阈值后，当 Context 的出方向吞吐量与出方向吞吐量限制值的比值超过了所设置的告警阈值，设备会生成告警日志；之后，当 Context 的出方向吞吐量与出方向吞吐量限制值的比值恢复到告警阈值以下，设备会生成恢复日志。
- 出方向吞吐量限速丢包日志功能：开启此功能后，当 Context 的出方向吞吐量达到出方向吞吐量限制值，设备会将超出限制值的报文丢弃，并对丢弃的报文生成日志信息；之后，如果该 Context 的出方向吞吐量降低到出方向吞吐量限制值以下，设备会生成恢复日志。

上面生成的日志信息将会被输出到信息中心模块处理，信息中心模块的配置将决定日志信息的发送规则和发送方向。有关信息中心的详细介绍，请参见“网络管理和监控配置指导”中的“信息中心”。

开启 Context 出方向吞吐量限速的 SNMP 告警功能后，上面生成的日志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 SNMP 的详细介绍，请参见“网络管理和监控配置指导”中的“SNMP”。

2. 配置限制和指导

因为此功能基于 CPU 进行吞吐量限制，所以其仅在非硬件快速转发的情况下生效。有关硬件快速转发功能的详细介绍，请参见“三层技术-IP 业务配置指导”中的“快速转发”。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 Context 视图。

context *context-name*

(3) 设置 Context 出方向的吞吐量限制。

capability throughput { kbps | pps } *value*

缺省情况下，各 Context 出方向不做吞吐量限制，按实际能力转发。

(4) （可选）开启 Context 出方向吞吐量告警功能并设置告警阈值。

context-capability throughput alarm enable alarm-threshold *alarm-threshold*

缺省情况下，Context 出方向吞吐量告警功能处于关闭状态。

(5) （可选）开启 Context 出方向吞吐量限速丢包日志功能。

context-capability throughput drop-logging enable

缺省情况下，Context 出方向吞吐量限速丢包日志功能处于关闭状态。

(6) （可选）开启 Context 出方向吞吐量限速的 SNMP 告警功能。

snmp-agent trap enable sib

缺省情况下，Context 出方向吞吐量限速的 SNMP 告警功能处于关闭状态。

1.7.2 限制 Context 对象策略规则总数

1. 功能简介

一个 Context 内可以配置多个对象策略，一个对象策略内包含多个规则。如果不加限制，会出现大量规则占用过多的内存的情况，影响 Context 的其它功能正常运行。所以，请根据需要为 Context 设置对象策略规则总数限制。当规则总数达到限制值时，后续不能新增规则。

一个 Context 的最大对象策略规则数，是在进驻的每个安全引擎内独立计算的，即 Context 进驻的每个安全引擎都有相同的对象策略规则数。

关于对象策略的详细描述请参见“安全配置指导”中的“对象策略”。

2. 配置限制和指导

如果设置的最大值比当前存在的规则总数小，配置仍会成功，多出的规则不会删除，依然生效，但不能新增规则。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 Context 视图。

context *context-name*

(3) 设置 Context 的对象策略规则总数限制。

capability object-policy-rule maximum *max-value*

缺省情况下，未对 Context 的对象策略规则总数进行限制。

1.7.3 限制 Context 安全策略规则总数

1. 功能简介

一个 Context 内可以配置多个安全策略规则。如果不加限制，会出现大量规则占用过多的内存的情况，影响 Context 的其它功能正常运行。所以，请根据需要为 Context 设置安全策略规则总数限制。当规则总数达到限制值时，后续不能新增规则。

一个 Context 的最大安全策略规则数，是在进驻的每个安全引擎内独立计算的，即 Context 进驻的每个安全引擎都有相同的安全策略规则数。

关于安全策略的详细描述请参见“安全配置指导”中的“安全策略”。

2. 配置限制和指导

如果设置的最大值比当前存在的规则总数小，配置仍会成功，多出的规则不会删除，依然生效，但不能新增规则。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 Context 视图。

context *context-name*

(3) 设置 Context 的安全策略规则总数限制。

capability security-policy-rule maximum *max-value*

缺省情况下，未对 Context 的安全策略规则总数进行限制。

1.7.4 限制 Context 会话并发数

1. 功能简介

如果一个 Context 建立了太多会话表会导致其他 Context 的会话由于内存不够而无法建立，为了防止这种情况，需要限制 Context 建立会话表的数量。

一个 Context 的最大会话数，是在进驻的每个安全引擎内独立计算的，即 Context 进驻的每个安全引擎都有相同的最大会话数，多个报文分散在不同引擎处理时，实际建立的会话数会大于限制的值。

Context 会话并发数限制对本机流量不生效，例如：FTP、Telnet、SSH、HTTP 和 HTTP 类型的七层负载均衡等业务。

2. 配置限制和指导

如果设置的最大值比当前存在的会话总数小，配置仍会成功，但不允许新建会话，且已经创建的会话不会被删除，依然生效。直到已建立的会话通过老化机制使得会话总数低于配置的最大值后，系统才允许新建会话。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 Context 视图。

context *context-name*

(3) 设置 Context 的单播会话并发数限制。

capability session maximum *max-number*

缺省情况下，未对 Context 允许的单播会话并发数进行限制。

1.7.5 限制 Context 会话新建速率

1. 功能简介

如果一个 Context 的会话新建速率过快会导致其他 Context 由于 CPU 处理能力不够而无法建立会话，为了防止这种情况，需要限制 Context 的会话新建速率。

一个 Context 的会话新建速率，是在进驻的每个安全引擎内独立计算的，即 Context 进驻的每个安全引擎都有相同的会话新建速率，多个报文分散在不同引擎处理时，实际的会话新建速率会大于限制的值。

Context 会话新建速率限制对本机流量不生效，例如：FTP、Telnet、SSH、HTTP 和 HTTP 类型的七层负载均衡等业务。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 Context 视图。

context *context-name*

(3) 设置 Context 的会话新建速率限制。

capability session rate *max-value*

缺省情况下，未对 Context 允许的会话新建速率进行限制。

1.7.6 限制 Context 的 SSL VPN 登录用户数

1. 功能简介

目前 SSL VPN 的用户数目由设备 License 控制，设备全部用户总数不能超过 License 控制，如果一个 Context 的用户总数到达了 License 限制，则会出现其他 Context 用户无法上线的问题，因此需要限制 Context 的上线用户数，同时设备全部用户总数仍受 License 控制。

2. 配置限制和指导

如果设置的数值小于当前 Context 的 SSL VPN 登录用户总数，则配置可以成功，但不再允许新的用户登录，且已经登录的用户不会被删除，依然生效。直到已登录的用户通过老化机制下线或用户主动下线，使得用户总数低于配置的最大值后，系统才允许新的用户登录。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 Context 视图。

context *context-name*

(3) 设置 Context 的 SSL VPN 登录用户数限制。

capability sslvpn-user maximum *max-number*

缺省情况下，未对 Context 的 SSL VPN 登录用户总数进行限制，由设备上 SSL VPN Licence 使用情况决定。

1.8 启动Context

1. 功能简介

Context 创建后需要启动，才能完成新 Context 的初始化，相当于上电启动。启动后，用户可以登录到该 Context 执行配置。

正常程序启动 Context 时，设备会先做一些检查（比如 Context 的主、备进程能否正常启动），满足条件后，才启动 Context，该命令会保证主备的 Context 状态一致，如果某成员设备或安全引擎上的 Context 启动失败，则会导致所有该 Context 进程启动失败。正常程序启动的 Context 能更好的保证 Context 的业务正常运行，所以，通常情况下，使用 **context start** 命令启动 Context 即可。**force** 参数用于以下场景：在 IRF 环境，如果主备倒换或者配置恢复过程中出现内存不足，会导致部分 Context 虽然可以处理业务，但因为它们的主、备进程状态不一致，这些 Context 一直停留在 updating 或者 inactive 状态。当内存资源恢复后，执行 context start force 命令，设备会在不中断业务的情况下，尽可能修复不正常的 Conext 进程，让这些 Context 恢复到正常状态。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 Context 视图。

context *context-name*

(3) 启动 Context。

context start [**force**]

1.9 为Context分配CPU/磁盘/内存资源

1.9.1 功能简介

缺省情况下，Context 会共享设备上的 CPU/磁盘/内存资源，为了防止一个 Context 过多的占用 CPU/磁盘/内存，而导致其它 Context 无法运行，需要限制 Context 对 CPU/磁盘/内存资源的使用。

1.9.2 为 Context 分配 CPU 权重

1. 功能简介

当 CPU 无法满足所有 Context 的处理需求时，系统将按照 CPU 权重值为每个 Context 分配处理时间。通过调整 Context 的权重，可以使指定的 Context 获得更多的 CPU 资源，保证关键业务的运行。例如：在三个 Context 中，将处理关键业务的 Context 的 CPU 权重设置为 2，其余两个 Context 的 CPU 权重设置为 1，则当 CPU 处理能力不足时，将为关键业务 Context 提供 2 倍于其它 Context 的处理时间。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 Context 视图。

context *context-name*

(3) 指定 Context 的 CPU 权重。

limit-resource cpu weight *weight-value*

缺省情况下，Context 的 CPU 权重为 10。

1.9.3 为 Context 分配磁盘空间上限

1. 配置限制和指导

建议在 Context 正常启动后再为 Context 分配磁盘空间上限，如果 Context 仅创建但未启动，那么磁盘使用值为 0，此时如果配置磁盘空间上限的值小于 Context 启动后正常实际使用的值，可能导致 Context 不能正常启动。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 Context 视图。

context *context-name*

(3) 显示 Context 对磁盘资源的使用情况。

display context resource disk

(4) 配置 Context 可使用的磁盘空间上限。

（独立运行模式）

limit-resource disk slot *slot-number* **cpu** *cpu-number* **ratio** *limit-ratio*

（IRF 模式）

limit-resource disk chassis *chassis-number* **slot** *slot-number* **cpu** *cpu-number* **ratio** *limit-ratio*

缺省情况下，进驻到同一安全引擎的所有 Context 共享该安全引擎的所有磁盘空间，每个 Context 可使用的磁盘空间上限为该安全引擎的空闲磁盘空间值。（独立运行模式）（IRF 模式）

如果设备上有多块磁盘，该命令对所有磁盘生效。

1.9.4 为 Context 分配内存空间上限

1. 配置限制和指导

建议在 Context 正常启动后再为 Context 分配内存空间上限，如果 Context 仅创建未启动，可能会由于内存不足，造成 Context 无法正常启动。在 Context 启动后，配置的内存上限值还不应过小，以免 Context 内业务申请不到内存后引起功能不正常。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 Context 视图。

context *context-name*

(3) 显示 Context 对内存资源的使用情况。

display context resource memory

(4) 配置 Context 可使用的内存空间上限。

（独立运行模式）

limit-resource memory slot *slot-number* **cpu** *cpu-number* **ratio** *limit-ratio*

（IRF 模式）

limit-resource memory chassis *chassis-number* **slot** *slot-number* **cpu** *cpu-number* **ratio** *limit-ratio*

缺省情况下，进驻到同一安全引擎的所有 Context 共享该安全引擎的所有内存空间，每个 Context 可使用的内存空间上限为该安全引擎的空闲内存空间值。（独立运行模式）（IRF 模式）

1.10 访问和管理Context

1. 功能简介

只要用户和设备之间路由可达，就能使用 **switchto context** 命令，通过设备和 Context 的内部连接，登录 Context。

除了上述方式，用户还可以通过 Context 上的接口，使用该 Context 的 IP 地址进行 Telnet/SSH 登录。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 登录 Context。

switchto context *context-name*

用户登录 Context 后，可以在 Context 的用户视图执行 **quit** 命令来退出登录。此时，命令视图将从当前 Context 的用户视图返回到缺省 Context 的系统视图。

1.11 配置Context限速功能

1.11.1 配置 Context 限制广播报文速率

1. 功能简介

如果一个 Context 接收和处理的广播报文过多，将会导致其他 Context 处理业务能力的下降，因此需要限制 Context 接收广播报文的数量。

Context 对入方向广播报文进行限速是通过整机接收报文限速和单个 Context 接收报文限速共同实现。当广播报文总速率和单个 Context 广播报文速率均达到各自阈值后，发往此 Context 的广播报文会被设备丢弃，否则不会被丢弃。

一个 Context 入方向广播报文的速率，在进驻的每个安全引擎内独立计算，即 Context 进驻的每个安全引擎都有相同的限速阈值，每个安全引擎对 Context 单独限制。当多个报文分散在不同安全引擎处理时，一个 Context 实际接收广播报文的速率可能大于设置的限速阈值。

2. 配置限制和指导

此功能仅限制入方向报文的速率。

此功能仅对使用共享接口且处于 Active 状态的 Context 生效。

当整机或单个 Context 广播限速阈值为零时表示不对广播报文限速。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 配置所有 Context 入方向广播报文的总速率限制。

context-capability inbound broadcast total pps *threshold*

缺省情况下，所有 Context 入方向广播报文总速率限制与设备型号有关，具体信息请参见命令参考手册。

(3) 配置缺省 Context 入方向广播报文的速率限制。

context-capability inbound broadcast single pps *threshold*

缺省情况下，缺省 Context 入方向广播报文限速速率为广播报文总速率阈值除以使用共享接口且处于 Active 状态的 Context 的总数。

(4) 进入 Context 视图。

context *context-name*

(5) 配置单个非缺省 Context 入方向广播报文的速率限制。

context-capability inbound broadcast single pps *threshold*

缺省情况下，单个非缺省 Context 入方向广播报文限速速率为广播报文总速率阈值除以使用共享接口且处于 Active 状态的 Context 的总数。

1.11.2 配置 Context 限制组播报文速率

1. 功能简介

如果一个 Context 接收和处理的组播报文过多，将会导致其他 Context 处理业务能力的下降，因此需要限制 Context 接收组播报文的数量。

Context 对入方向组播报文进行限速是通过整机接收报文限速和单个 Context 接收报文限速共同实现。当组播报文总速率和单个 Context 组播报文速率均达到各自阈值后，发往此 Context 的组播报文会被设备丢弃，否则不会被丢弃。

一个 Context 入方向组播报文的速率，在进驻的每个安全引擎内独立计算，即 Context 进驻的每个安全引擎都有相同的限速阈值，每个安全引擎对 Context 单独限制。当多个报文分散在不同安全引擎处理时，一个 Context 实际接收组播报文的速率可能大于设置的限速阈值。

2. 配置限制和指导

此功能仅限制入方向报文的速率。

此功能仅对使用共享接口且处于 Active 状态的 Context 生效。

当整机或单个 Context 组播限速阈值为零时表示不对组播报文进行限速。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 配置所有 Context 入方向组播报文的总速率限制。

context-capability inbound multicast total pps *threshold*

缺省情况下，所有 Context 入方向组播报文总速率限制与设备型号有关，具体信息请参见命令参考手册。

(3) 配置缺省 Context 入方向组播报文的速率限制。

context-capability inbound multicast single pps *threshold*

缺省情况下，缺省 Context 入方向组播报文限速速率为组播报文总速率阈值除以使用共享接口且处于 Active 状态的 Context 的总数。

(4) 进入 Context 视图。

context *context-name*

(5) 配置单个非缺省 Context 入方向组播报文的速率限制。

context-capability inbound multicast single pps *threshold*

缺省情况下，单个非缺省 Context 入方向组播报文限速速率为组播报文总速率阈值除以使用共享接口且处于 Active 状态的 Context 的总数。

1.11.3 开启 Context 限速丢包日志功能

1. 功能简介

开启此功能后，当 Context 接收到的广播报文或组播报文因达到系统设置的阈值而被丢弃时，设备将会对丢弃的报文生成日志信息。此日志信息将会被输出到信息中心模块处理，信息中心模块的配置将决定日志信息的发送规则和发送方向。有关信息中心的详细介绍，请参见“网络管理和监控配置指导”中的“信息中心”。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 开启 Context 入方向报文限速丢包日志功能。

context-capability inbound drop-logging enable

缺省情况下，Context 入方向报文限速丢包日志功能处于开启状态。

1.12 配置CPU核的攻击防范阈值

1. 功能简介

此功能对所有 Context 入方向上的所有报文（包括广播报文、组播报文和单播报文）均生效。

当某 CPU 核的利用率达到此攻击防范阈值，并且驱动公共队列已满时，系统则认为该 CPU 核受到了攻击。这时，系统将按照配置的单核 CPU 攻击防范动作（通过 **attack-defense cpu-core action** 命令配置）对报文进行相应的处理。

有关 **attack-defense cpu-core action** 命令详细介绍，请参见“安全命令参考”中的“攻击检测与防范”。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 配置设备入方向所有报文对 CPU 核的攻击防范阈值。

context-capability inbound unicast total cpu-usage *threshold*

缺省情况下，设备入方向所有报文对 CPU 核的攻击防范阈值为 95%。

1.13 收集各Context的日志信息

1. 功能简介

此功能可以收集 logfile 文件夹和 diagfile 文件夹下的所有文件。

2. 配置步骤

请在用户视图下执行本命令，收集各 Context 的日志信息

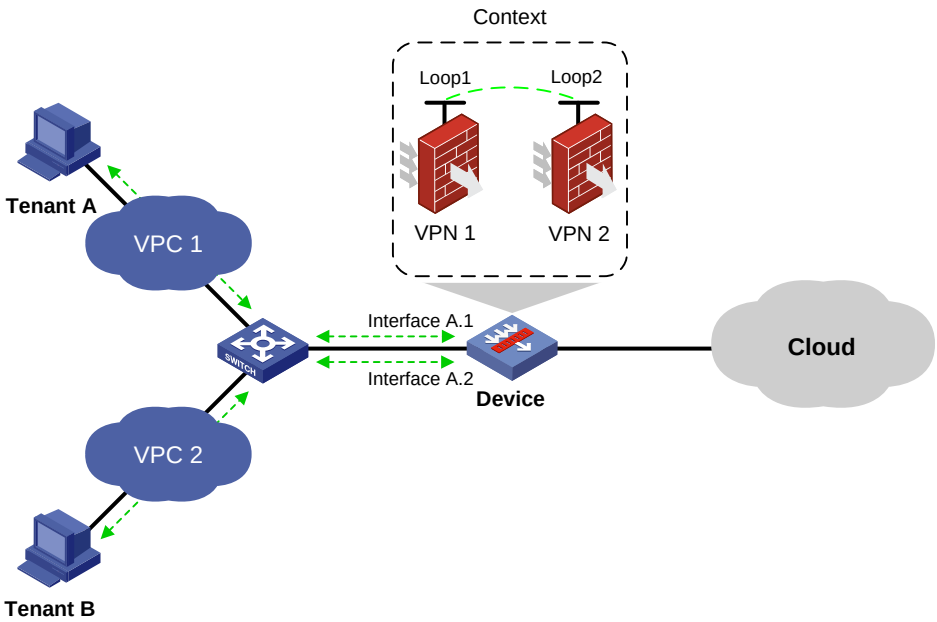
tar context [**name** *context-name*] **log file** *filename*

1.14 配置Context支持跨VPC流量互通

1. 功能简介

在 VPC（Virtual Private Cloud，虚拟私有云）场景中，不同租户被不同的 VPC 隔离。设备上使用 VPN 实例与 VPC 进行一一对应，这样既可以保证不同 VPC 租户之间流量的隔离，又可以实现不同 VPC 租户之间流量的互通。

图1-2 Context 中跨 VPC 流量互通示意图



如图 1-2 所示，不同 VPC 之间流量隔离和互通的实现机制如下：

- 在设备上创建不同的 VPN 实例用于区分和隔离 VPC 之间的流量。

- 在设备上为 VPN 实例配置静态路由，将路由出接口配置为 LoopBack 接口，可实现 VPC 间流量的互通。

2. 配置限制和指导

跨 VPN 实例转发流量只支持类似安全策略这种报文过滤和阻断的业务，仅支持静态路由配置。

推荐使用设备的物理子接口或者逻辑子接口与 VPC 租户进行连接。

开启跨 VPC 的硬件快速转发功能与 GRE 和 MPLS 功能互斥不能同时使用。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 创建 VPN 实例，并与接口关联。

有关创建 VPN 实例并与接口关联的具体操作，请参见“MPLS 配置指导”中的“MCE”。

(3) 为 VPN 实例配置静态路由，路由出接口为 LoopBack 接口。

有关为 VPN 实例配置静态路由的具体操作，请参见“三层技术-IP 路由配置指导”中的“静态路由”和“IPv6 静态路由”。

(4) 开启跨 VPC 的硬件快速转发功能。

（独立运行模式）

hardware fast-forwarding vpc enable [**slot** *slot-number* [**cpu** *cpu-number*]]

（IRF 模式）

hardware fast-forwarding vpc enable [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]]

缺省情况下，跨 VPC 的硬件快速转发功能处于关闭状态。

1.15 Context显示和维护

在完成 Context 相关配置后，在任意视图下执行 **display** 命令，可以显示配置后 Context 的运行情况，通过查看显示信息，来验证配置的效果。

在用户视图下，用户可以执行 **reset** 命令清除 Context 相关信息。

表1-1 缺省 Context 上可执行的显示和维护

操作	命令
显示安全引擎组的信息	**display blade-controller-team** [*blade-controller-team-name* | **id** *blade-controller-team-id*]
显示Context的相关信息	**display context** [**name** *context-name*] [**verbose**]

操作	命令
显示Context内可分配业务资源的使用情况	（独立运行模式） **display context** [**name** *context-name*] **capability** [**security-policy** | **session** [**slot** *slot-number* **cpu** *cpu-number*] | **sslvpn-user**] （IRF模式） **display context** [**name** *context-name*] **capability** [**security-policy** |**session** [**chassis** *chassis-number* **slot** *slot-number* **cpu** *cpu-number*] | **sslvpn-user**]
显示Context入方向广播报文的速率限制的统计信息	（独立运行模式） **display context name** *context-name* **capability inbound broadcast slot** *slot-number* **cpu** *cpu-number* （IRF模式） **display context name** *context-name* **capability inbound broadcast chassis** *chassis-number* **slot** *slot-number* **cpu** *cpu-number*
显示Context入方向组播报文的速率限制的统计信息	（独立运行模式） **display context name** *context-name* **capability inbound multicast slot** *slot-number* **cpu** *cpu-number* （IRF模式） **display context name** *context-name* **capability inbound multicast chassis** *chassis-number* **slot** *slot-number* **cpu** *cpu-number*
显示CPU核受到攻击的相关统计信息	（独立运行模式） **display capability inbound unicast slot** *slot-number* **cpu** *cpu-number* （IRF模式） **display capability inbound unicast chassis** *chassis-number* **slot** *slot-number* **cpu** *cpu-number*
显示各Context的配置信息	**display context** [**name** *context-name*] **configuration** [**file** *filename*]
显示Context的接口列表	**display context** [**name** *context-name*] **interface**
显示Context对CPU/磁盘/内存资源的使用情况	（独立运行模式） **display context** [**name** *context-name*] **resource** [**cpu** | **disk** | **memory**] [**slot** *slot-number* **cpu** *cpu-number*] （IRF模式） **display context** [**name** *context-name*] **resource** [**cpu** | **disk** | **memory**] [**chassis** *chassis-number* **slot** *slot-number* **cpu** *cpu-number*]
显示Context内资源的统计信息	**display context** [**name** *context-name*] **statistics** [**file** *filename*]
显示Context的VLAN列表	**display context** [**name** *context-name*] **vlan**
显示非缺省Context的重启信息	**display context name** *context-name* **reboot** *show-number* [*offset*]
显示所有Context内SSL VPN在线用户数	**display context online-users sslvpn**

操作	命令
清除指定安全引擎组中不在位的安全引擎的数据信息	（独立运行模式） **reset blade-controller-team** *team-id* **member slot** *slot-number* **cpu** *cpu-number* （IRF模式） **reset blade-controller-team** *team-id* **member chassis** *chassis-number* **slot** *slot-number* **cpu** *cpu-number*
清除Context入方向广播报文的速率限制的统计信息	（独立运行模式） **reset context name** *context-name* **capability inbound broadcast slot** *slot-number* **cpu** *cpu-number* （IRF模式） **reset context name** *context-name* **capability inbound broadcast chassis** *chassis-number* **slot** *slot-number* **cpu** *cpu-number*
清除Context入方向组播报文的速率限制的统计信息	（独立运行模式） **reset context name** *context-name* **capability inbound multicast slot** *slot-number* **cpu** *cpu-number* （IRF模式） **reset context name** *context-name* **capability inbound multicast chassis** *chassis-number* **slot** *slot-number* **cpu** *cpu-number*
清除非缺省Context的重启信息	**reset context** [**name** *context-name*] **reboot**

表1-2 非缺省 Context 上可执行的显示和维护

操作	命令
显示Context的接口列表	**display context** [**name** *context-name*] **interface**
显示本Context的重启信息	**display context reboot** *show-number* [*offset*]
清除本Context的重启信息	**reset context reboot**

1.16 Context典型配置举例

1.16.1 Context 基本组网配置举例（独立运行模式）

1. 组网需求

将设备 Device 虚拟成三台独立的 Device：Context cnt1、Context cnt2、Context cnt3，并分给三个不同的用户网络进行安全防护。要求在用户侧看来，各自的接入设备是独享的。

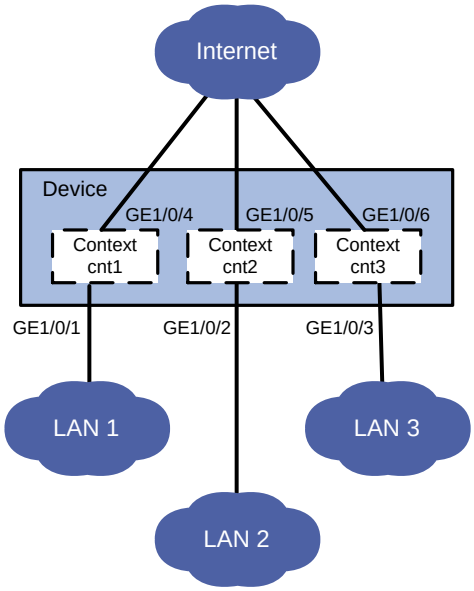
- LAN 1、LAN 2、LAN 3 分别属于公司 A、公司 B、公司 C，现各公司的网络均需要进行安全防护。公司 A 使用的网段为 192.168.1.0/24，公司 B 使用的网段为 192.168.2.0/24，公司 C 使用的网段为 192.168.3.0/24。
- 公司 A 的用户多，业务需求复杂，因此需要给 Context cnt1 提供较大的磁盘/内存空间使用上限，以便保存配置文件、启动文件和系统信息等；对公司 B 使用系统缺省的磁盘空间即可；

公司 C 人员规模小，上网流量比较少，对接入 Device 的配置及性能要求较低，因此对 Context cnt3 提供较低的 CPU 权重。

- GigabitEthernet1/0/1 和 GigabitEthernet1/0/4 分配给 Context cnt1、GigabitEthernet1/0/2 和 GigabitEthernet1/0/5 分配给 Context cnt2、GigabitEthernet1/0/3 和 GigabitEthernet1/0/6 分配给 Context cnt3。

2. 组网图

图1-3 Context 基本组网配置组网图



3. 配置步骤

(1) 配置安全引擎组 test

创建安全引擎组 test，并将 3 号槽位 cpu 号为 1 的安全引擎加入该安全引擎组。

```
<Device> system-view
[Device] blade-controller-team test
[Device-blade-controller-team-2-test] location blade-controller slot 3 cpu 1
This operation will also reboot the blade controller. Continue? [Y/N]:y
[Device-blade-controller-team-2-test] quit
```

(2) 创建并配置 Context cnt1，供公司 A 使用

创建 Context cnt1，进驻安全引擎组，配置其磁盘和内存使用的上限均为 60%、CPU 权重为 8，具体配置步骤如下。

```
[Device] context cnt1
[Device-context-2-cnt1] description context-1
[Device-context-2-cnt1] location blade-controller-team 2
[Device-context-2-cnt1] limit-resource disk slot 3 cpu 1 ratio 60
[Device-context-2-cnt1] limit-resource memory slot 3 cpu 1 ratio 60
[Device-context-2-cnt1] limit-resource cpu weight 8
```

将接口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/4 分配给 Context cnt1。

```
[Device-context-2-cnt1] allocate interface gigabitethernet 1/0/1 gigabitethernet 1/0/4
```

启动 Context cnt1。

```
[Device-context-2-cnt1] context start
It will take some time to start the context...
Context started successfully.
[Device-context-2-cnt1] quit
```

切换到 Context cnt1。

```
[Device] switchto context cnt1
******************************************************************************
* Copyright (c) 2004-2021 New H3C Technologies Co., Ltd. All rights reserved.*
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************


<H3C> system-view
```

配置 Telnet 功能，保证管理用户可以正常登录设备，具体配置步骤请参考“基础配置指导”中的“登录设备”。

将 Context cnt1 的名称修改为 cnt1，以便和其它 Context 区别。

```
[H3C] sysname cnt1
```

配置接口 GigabitEthernet1/0/1 的 IP 地址为 192.168.1.251，供公司 A 的管理用户远程登录。

```
[cnt1] interface gigabitethernet 1/0/1
[cnt1-GigabitEthernet1/0/1] ip address 192.168.1.251 24
```

从自定义 Context cnt1 返回缺省 Context。

```
[cnt1-GigabitEthernet1/0/1] return
<cnt1> quit
[Device]
```

(3) 创建并配置 Context cnt2，供公司 B 使用

创建 Context cnt2，设置描述信息

```
[Device] context cnt2
[Device-context-3-cnt2] description context-2
```

设置 cnt2 进驻安全引擎组 test（编号为 2）。

```
[Device-context-3-cnt2] location blade-controller-team 2
```

将接口 GigabitEthernet1/0/2 和 GigabitEthernet1/0/5 分配给 Context cnt2。

```
[Device-context-3-cnt2] allocate interface gigabitethernet 1/0/2 gigabitethernet 1/0/5
```

启动 Context cnt2。

```
[Device-context-3-cnt2] context start
```

```
It will take some time to start the context...
Context started successfully.
[Device-context-3-cnt2] quit
```

切换到 Context cnt2。

```
[Device] switchto context cnt2
******************************************************************************
* Copyright (c) 2004-2021 New H3C Technologies Co., Ltd. All rights reserved.*
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************


<H3C> system-view
```

配置 Telnet 功能，保证管理用户可以正常登录设备，具体配置步骤请参考“基础配置指导”中的“登录设备”。

将 Context cnt2 的名称修改为 cnt2，以便和其它 Context 区别。

```
[H3C] sysname cnt2
```

配置接口 GigabitEthernet1/0/2 的 IP 地址为 192.168.2.251，供公司 B 的管理用户远程登录。

```
[cnt2] interface gigabitethernet 1/0/2
[cnt2-GigabitEthernet1/0/2] ip address 192.168.2.251 24
```

从自定义 Context cnt2 返回缺省 Context。

```
[cnt2-GigabitEthernet1/0/2] return
<cnt2> quit
[Device]
```

(4) 创建并配置 Context cnt3，供公司 C 使用

创建 Context cnt3，设置描述信息

```
[Device] context cnt3
[Device-context-4-cnt3] description context-3
```

设置 cnt3 进驻安全引擎组 test（编号为 2）。

```
[Device-context-4-cnt3] location blade-controller-team 2
```

配置 Context cnt3 的 CPU 权重为 2。

```
[Device-context-4-cnt3] limit-resource cpu weight 2
```

将接口 GigabitEthernet1/0/3 和 GigabitEthernet1/0/6 分配给 Context cnt3。

```
[Device-context-4-cnt3] allocate interface gigabitethernet 1/0/3 gigabitethernet 1/0/6
```

启动 Context cnt3。

```
[Device-context-4-cnt3] context start
It will take some time to start the context...
Context started successfully.
```

```
[Device-context-4-cnt3] quit
```

切换到 Context cnt3。

```
[Device] switchto context cnt3
******************************************************************************
* Copyright (c) 2004-2021 New H3C Technologies Co., Ltd. All rights reserved.*
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************


<H3C> system-view
```

配置 Telnet 功能，保证管理用户可以正常登录设备，具体配置步骤请参考“基础配置指导”中的“登录设备”。

将 Context cnt3 的名称修改为 cnt3，以便和其它 Context 区别。

```
[H3C] sysname cnt3
```

配置接口 GigabitEthernet1/0/3 的 IP 地址为 192.168.3.251，供公司 C 的管理用户远程登录。

```
[cnt3] interface gigabitethernet 1/0/3
[cnt3-GigabitEthernet1/0/3] ip address 192.168.3.251 24
```

从自定义 Context cnt3 返回缺省 Context。

```
[cnt3-GigabitEthernet1/0/3] return
<cnt3> quit
[Device]
```

4. 验证配置

(1) 查看 Context 是否存在并且运转正常。（此时，Device 上应该有四台处于正常工作 active 状态的 Context）

```
[Device] display context
ID     Name             Status       Description
1      Admin            active       DefaultContext
2      cnt1             active       context-1
3      cnt2             active       context-2
4      cnt3             active       context-3
```

(2) 模拟公司 A 的管理用户登录到 Context cnt1，可以查看本设备的当前配置。

```
C:\> telnet 192.168.1.251
******************************************************************************
* Copyright (c) 2004-2021 New H3C Technologies Co., Ltd. All rights reserved.*
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************
```

```
<cnt1> display current-configuration
......
```

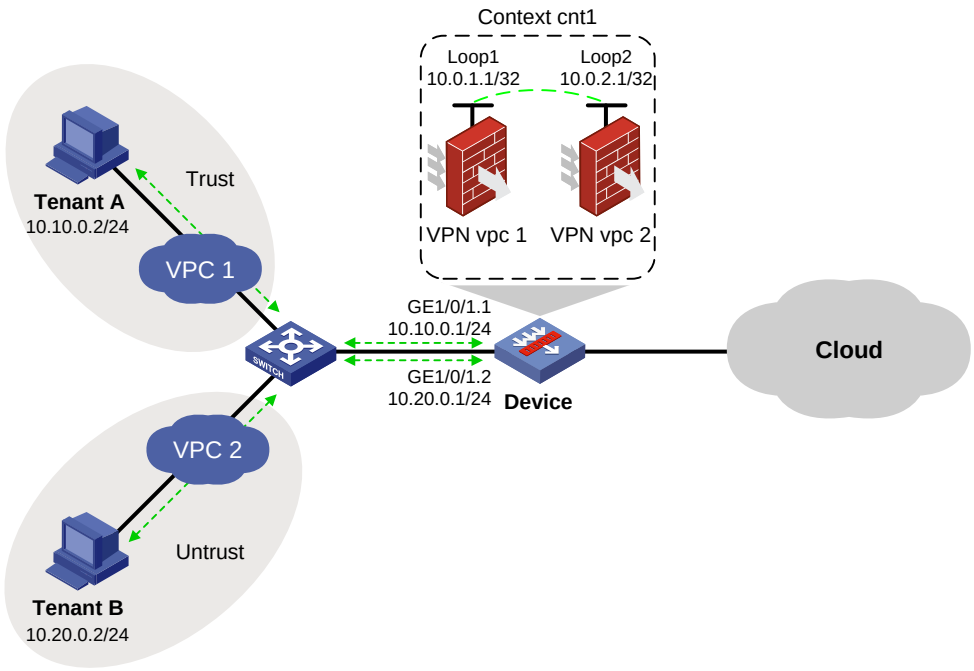
1.16.2 Context 支持跨 VPC 流量互通典型配置举例（独立运行模式）

1. 组网需求

设备上的非缺省 Context 在对 VPC 租户访问公有云的流量进行安全防护的同时，也可以使不同 VPC 之间的流量安全互通。

2. 组网图

图1-4 Context 支持跨 VPC 流量互通配置组网图



3. 配置步骤

(1) 创建并配置 Context cnt1，供 VPC1 和 VPC2 使用。

创建名称为 cnt1 的 Context，并为其分配 CPU、内存、磁盘和接口资源，其具体配置步骤请参见“1.16.1 Context 支持跨 VPC 流量互通典型配置举例（独立运行模式）”中的相关内容，本举例不再赘述。

(2) 切换到 Context cnt1。

```
<Device> system-view
[Device] switchto context cnt1
******************************************************************************
* Copyright (c) 2004-2021 New H3C Technologies Co., Ltd. All rights reserved.*
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
```

```
**************************************************************************


<H3C> system-view
```

配置 Telnet 功能，保证管理用户可以正常登录设备，具体配置步骤请参考“基础配置指导”中的“登录设备”。

将 Context cnt1 的名称修改为 cnt1，以便和其它 Context 区别。

```
[H3C] sysname cnt1
```

(3) 创建 VPN 实例，并将接口关联 VPN 实例。

请根据组网中规划的信息，创建 VPN 实例、LoopBack 接口和以太网子接口，在以太网子接口上终结最外层 VLAN ID，具体配置步骤如下。

```
[cnt1] ip vpn-instance vpc1
[cnt1-vpn-instance-vpc1] quit
[cnt1] ip vpn-instance vpc2
[cnt1-vpn-instance-vpc2] quit
[cnt1] interface loopback 1
[cnt1-LoopBack1] ip binding vpn-instance vpc1
[cnt1-LoopBack1] ip address 10.0.1.1 255.255.255.255
[cnt1-LoopBack1] quit
[cnt1] interface loopback 2
[cnt1-LoopBack2] ip binding vpn-instance vpc2
[cnt1-LoopBack2] ip address 10.0.2.1 255.255.255.255
[cnt1-LoopBack2] quit
[cnt1] interface gigabitethernet 1/0/1.1
[cnt1-GigabitEthernet1/0/1.1] ip binding vpn-instance vpc1
[cnt1-GigabitEthernet1/0/1.1] ip address 10.10.0.1 255.255.255.0
[cnt1-GigabitEthernet1/0/1.1] vlan-type dot1q vid 10
[cnt1-GigabitEthernet1/0/1.1] quit
[cnt1] interface gigabitethernet 1/0/1.2
[cnt1-GigabitEthernet1/0/1.2] ip binding vpn-instance vpc2
[cnt1-GigabitEthernet1/0/1.2] ip address 10.20.0.1 255.255.255.0
[cnt1-GigabitEthernet1/0/1.2] vlan-type dot1q vid 20
[cnt1-GigabitEthernet1/0/1.2] quit
```

(4) 配置静态路由，保证路由可达

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由使 VPC 1 和 VPC 2 之间路由可达，具体配置步骤如下。

```
[cnt1] ip route-static vpn-instance vpc1 10.20.0.0 24 loopback1 10.0.2.1
[cnt1] ip route-static vpn-instance vpc2 10.10.0.0 24 loopback2 10.0.1.1
```

(5) 开启跨 VPC 的硬件快速转发功能。

开启跨 VPC 的硬件快速转发功能提高报文转发速度。

```
[cnt1] hardware fast-forwarding vpc enable
```

(6) 配置安全域。

将 LoopBack 接口和以太网子接口加入安全域。

```
[cnt1] security-zone name trust
[cnt1-security-zone-Trust] import interface loopback1
[cnt1-security-zone-Trust] import interface gigabitethernet 1/0/1.1
[cnt1-security-zone-Trust] quit
[cnt1] security-zone name untrust
[cnt1-security-zone-Untrust] import interface loopback2
[cnt1-security-zone-Untrust] import interface gigabitethernet 1/0/1.2
[cnt1-security-zone-Untrust] quit
```

(7) 配置安全策略保证 VPC 之间的流量互通。

```
[cnt1] security-policy ip
[cnt1-security-policy-ip] rule name vpc1
[cnt1-security-policy-ip-0-vpc1] action pass
[cnt1-security-policy-ip-0-vpc1] vrf vpc1
[cnt1-security-policy-ip-0-vpc1] source-zone trust
[cnt1-security-policy-ip-0-vpc1] destination-zone untrust
[cnt1-security-policy-ip-0-vpc1] quit
[cnt1-security-policy-ip] rule name vpc2
[cnt1-security-policy-ip-1-vpc2] action pass
[cnt1-security-policy-ip-1-vpc2] vrf vpc2
[cnt1-security-policy-ip-1-vpc2] source-zone untrust
[cnt1-security-policy-ip-1-vpc2] destination-zone trust
[cnt1-security-policy-ip-1-vpc2] quit
[cnt1-security-policy-ip] quit
```

4. 验证配置

在租户 A 上可以 Ping 通租户 B。

```
C:\> ping 10.20.0.2

Pinging 10.20.0.2 with 32 bytes of data:

Reply from 10.20.0.2: bytes=32 time=19ms TTL=254
Reply from 10.20.0.2: bytes=32 time<1ms TTL=254
Reply from 10.20.0.2: bytes=32 time<1ms TTL=254
Reply from 10.20.0.2: bytes=32 time<1ms TTL=254
```

```
Ping statistics for 10.20.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 （0% loss），
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 19ms, Average = 4ms
```

1.16.3 通过 Context 实现云计算中心网关配置举例（独立运行模式）

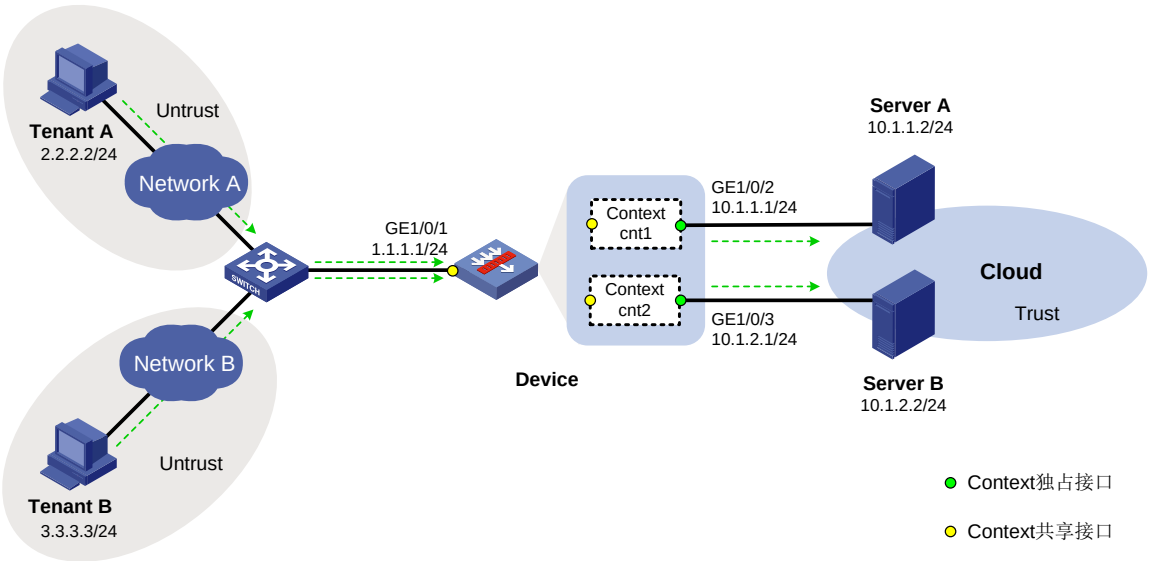
1. 组网需求

设备 Device 作为云计算中心的出口网关，对内部网络的信息安全进行防护，其中 Device 只有一个公网接口 GigabitEthernet1/0/1。现有相互独立的租户 A 和 B 需要使用云计算中心的计算资源，具体组网需求如下：

- 将 Device 虚拟成两台独立的 Device：Context cnt1 和 Context cnt2，并分给租户 A 和 B 进行安全防护。其中，Context cnt1 以共享方式使用接口 GigabitEthernet1/0/1，以独占方式使用接口 GigabitEthernet1/0/2；Context cnt2 以共享方式使用接口 GigabitEthernet1/0/1，以独占方式使用接口 GigabitEthernet1/0/3。
- 在共享接口 GigabitEthernet1/0/1 上配置 NAT 内部服务器，使租户 A 和 B 可以利用独立的公网 IP 地址访问 Server A 和 Server B。

2. 组网图

图1-5 通过 Context 实现云计算中心网关配置组网图



3. 配置步骤

(1) 配置接口 IP 地址

根据组网图中规划的信息，配置接口 GigabitEthernet1/0/1 的 IP 地址，具体配置步骤如下。

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
```

```
[Device-GigabitEthernet1/0/1] ip address 1.1.1.1 24
[Device-GigabitEthernet1/0/1] quit
```

(2) 配置安全引擎组 test

创建安全引擎组 test，并将 3 号槽位 cpu 号为 1 的安全引擎加入该安全引擎组。

```
[Device] blade-controller-team test
[Device-blade-controller-team-2-test] location blade-controller slot 3 cpu 1
This operation will also reboot the blade controller. Continue? [Y/N]:y
[Device-blade-controller-team-2-test] quit
```

(3) 创建并配置 Context cnt1，供租户 A 使用

创建 Context cnt1，设置描述信息。

```
[Device] context cnt1
[Device-context-2-cnt1] description context-1
```

设置 cnt1 进驻安全引擎组 test（编号为 2）。

```
[Device-context-2-cnt1] location blade-controller-team 2
```

以共享方式将接口 GigabitEthernet1/0/1 分配给 Context cnt1。

```
[Device-context-2-cnt1] allocate interface gigabitethernet 1/0/1 share
```

以独占方式将接口 GigabitEthernet1/0/2 分配给 Context cnt1。

```
[Device-context-2-cnt1] allocate interface gigabitethernet 1/0/2
Configuration of the interfaces will be lost. Continue? [Y/N]:y
```

启动 Context cnt1。

```
[Device-context-2-cnt1] context start
It will take some time to start the context...
Context started successfully.
[Device-context-2-cnt1] quit
```

切换到 Context cnt1。

```
[Device] switchto context cnt1
******************************************************************************
* Copyright (c) 2004-2021 New H3C Technologies Co., Ltd. All rights reserved.*
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************


<H3C> system-view
```

将 Context cnt1 的名称修改为 cnt1，以便和其它 Context 区别。

```
[H3C] sysname cnt1
```

配置接口 GigabitEthernet1/0/2 的 IP 地址为 10.1.1.1/24。

```
[cnt1] interface gigabitethernet 1/0/2
[cnt1-GigabitEthernet1/0/2] ip address 10.1.1.1 24
```

```
[cnt1-GigabitEthernet1/0/2] quit
```

将接口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/2 分别加入 Untrust 和 Trust 安全域。

```
[cnt1] security-zone name untrust
[cnt1-security-zone-Untrust] import interface gigabitethernet 1/0/1
[cnt1-security-zone-Untrust] quit
[cnt1] security-zone name trust
[cnt1-security-zone-Trust] import interface gigabitethernet 1/0/2
[cnt1-security-zone-Trust] quit
```

配置安全策略保证租户 A 能成功访问 Server A。

```
[cnt1] security-policy ip
[cnt1-security-policy-ip] rule name untrust-trust
[cnt1-security-policy-ip-0-untrust-trust] action pass
[cnt1-security-policy-ip-0-untrust-trust] source-zone untrust
[cnt1-security-policy-ip-0-untrust-trust] destination-zone trust
[cnt1-security-policy-ip-0-untrust-trust] source-ip-host 2.2.2.2
[cnt1-security-policy-ip-0-untrust-trust] destination-ip-host 10.1.1.2
[cnt1-security-policy-ip-0-untrust-trust] quit
[cnt1-security-policy-ip] quit
```

从自定义 Context cnt1 返回缺省 Context。

```
[cnt1] quit
<cnt1> quit
[Device]
```

(4) 创建并配置 Context cnt2，供租户 B 使用

创建 Context cnt2，设置描述信息。

```
[Device] context cnt2
[Device-context-3-cnt2] description context-2
```

设置 cnt2 进驻安全引擎组 test（编号为 2）。

```
[Device-context-3-cnt2] location blade-controller-team 2
```

以共享方式将接口 GigabitEthernet1/0/1 分配给 Context cnt2。

```
[Device-context-3-cnt2] allocate interface gigabitethernet 1/0/1 share
```

以独占方式将接口 GigabitEthernet1/0/3 分配给 Context cnt2。

```
[Device-context-3-cnt2] allocate interface gigabitethernet 1/0/3
Configuration of the interfaces will be lost. Continue? [Y/N]:y
```

启动 Context cnt2。

```
[Device-context-3-cnt2] context start
It will take some time to start the context...
Context started successfully.
[Device-context-3-cnt2] quit
```

切换到 Context cnt2。

```
[Device] switchto context cnt2
******************************************************************************
* Copyright (c) 2004-2021 New H3C Technologies Co., Ltd. All rights reserved.*
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************


<H3C> system-view
```

将 Context cnt2 的名称修改为 cnt2，以便和其它 Context 区别。

```
[H3C] sysname cnt2
```

配置接口 GigabitEthernet1/0/3 的 IP 地址为 10.1.2.1/24。

```
[cnt2] interface gigabitethernet 1/0/3
[cnt2-GigabitEthernet1/0/3] ip address 10.1.2.1 24
[cnt2-GigabitEthernet1/0/3] quit
```

将接口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/3 分别加入 Untrust 和 Trust 安全域。

```
[cnt2] security-zone name untrust
[cnt2-security-zone-Untrust] import interface gigabitethernet 1/0/1
[cnt2-security-zone-Untrust] quit
[cnt2] security-zone name trust
[cnt2-security-zone-Trust] import interface gigabitethernet 1/0/3
[cnt2-security-zone-Trust] quit
```

配置安全策略保证租户 B 能成功访问 Server B。

```
[cnt2] security-policy ip
[cnt2-security-policy-ip] rule name untrust-trust
[cnt2-security-policy-ip-0-untrust-trust] action pass
[cnt2-security-policy-ip-0-untrust-trust] source-zone untrust
[cnt2-security-policy-ip-0-untrust-trust] destination-zone trust
[cnt2-security-policy-ip-0-untrust-trust] source-ip-host 3.3.3.3
[cnt2-security-policy-ip-0-untrust-trust] destination-ip-host 10.1.2.2
[cnt2-security-policy-ip-0-untrust-trust] quit
[cnt2-security-policy-ip] quit
```

从自定义 Context cnt2 返回缺省 Context。

```
[cnt2] quit
<cnt2> quit
[Device]
```

(5) 配置 NAT 内部服务器。

在接口 GigabitEthernet1/0/1 上配置 NAT 内部服务器，允许外部通过 http://1.1.1.2:8080 地址访问 Server A，通过 http://1.1.1.3:8080 地址访问 Server B。

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] nat server protocol tcp global 1.1.1.2 8080 inside 10.1.1.2 http
[Device-GigabitEthernet1/0/1] nat server protocol tcp global 1.1.1.3 8080 inside 10.1.2.2 http
[Device-GigabitEthernet1/0/1] quit
```

4. 验证配置

(1) 查看 Context 是否存在并且运转正常。（此时，Device 上应该有三台处于正常工作 active 状态的 Context）

```
[Device] display context
ID     Name             Status       Description
1      Admin            active       DefaultContext
2      cnt1             active       context-1
3      cnt2             active       context-2
```

(2) 租户 A 可以通过 http://1.1.1.2:8080 地址访问 Server A，租户 B 可以通过 http://1.1.1.3:8080 地址访问 Server B。

目 录

1 以太网冗余接口 ····· 1-1
1.1 以太网冗余接口简介 ····· 1-1
1.1.1 以太网冗余接口的工作原理 ····· 1-1
1.1.2 以太网冗余接口的应用场景 ····· 1-1
1.1.3 以太网冗余子接口 ····· 1-1
1.2 配置以太网冗余接口 ····· 1-2
1.2.1 配置限制和指导 ····· 1-2
1.2.2 配置以太网冗余接口的基本参数 ····· 1-2
1.2.3 配置激活状态通告参数 ····· 1-3
1.2.4 开启以太网冗余接口的流量快速切换功能 ····· 1-4
1.3 配置以太网冗余子接口 ····· 1-4
1.3.2 恢复当前以太网冗余接口/以太网冗余子接口的缺省配置 ····· 1-5
1.4 以太网冗余接口显示和维护 ····· 1-6
2 冗余组 ····· 2-1
2.1 冗余组简介 ····· 2-1
2.1.1 冗余组工作原理 ····· 2-1
2.1.2 冗余组节点的状态 ····· 2-2
2.1.3 冗余组成员 ····· 2-2
2.1.4 冗余组的倒换/倒回机制 ····· 2-5
2.2 冗余组配置任务简介 ····· 2-6
2.3 创建冗余组 ····· 2-6
2.4 配置冗余组节点 ····· 2-6
2.5 将物理以太网接口加入冗余组 ····· 2-7
2.6 将以太网冗余接口加入冗余组 ····· 2-8
2.7 将备份组加入冗余组 ····· 2-8
2.8 配置冗余组定时器 ····· 2-9
2.9 手工触发冗余组倒换 ····· 2-9
2.10 手工触发冗余组倒回 ····· 2-9
2.11 开启冗余组告警功能 ····· 2-10
2.12 冗余组显示和维护 ····· 2-10
2.13 冗余组典型配置举例 ····· 2-10
2.13.1 工作在三层，上下行分别连接两台路由器，两台路由器接口不在同一网段 ····· 2-10

2.13.2 工作在三层，上下行分别连接一台设备 ·····2-17

1 以太网冗余接口

1.1 以太网冗余接口简介

以太网冗余接口（Redundant Ethernet，Reth）是一种三层虚拟接口。一个以太网冗余接口中包含两个成员接口，使用以太网冗余接口可以实现这两个接口之间的冗余备份。

说明

仅 IRF 模式支持以太网冗余接口功能。

1.1.1 以太网冗余接口的工作原理

以太网冗余接口的成员接口有两种状态：

- 激活状态：能够收发报文。
- 非激活状态：不能收发报文。

任意时刻，同一个以太网冗余接口内只有一个成员接口处于激活状态。

当两个成员接口的物理状态均为 up 时，优先级较高的成员接口处于激活状态，优先级较低的成员接口处于非激活状态。优先级可通过命令行配置。

当激活接口的链路变为 down 时，处于非激活状态的接口会自动激活，接替原激活接口收发报文，实现接口间的备份。

在上、下行设备看来，与其连接的是以太网冗余接口，学习到的是以太网冗余接口的 MAC 地址。成员接口的激活状态发生变化，不会影响上、下行设备。

说明

如果以太网冗余接口加入了冗余组，由冗余组决定哪个成员接口处于激活状态。具体描述请参见“2 冗余组”。

1.1.2 以太网冗余接口的应用场景

以太网冗余接口通常和冗余组配合使用，具体情况请参见“虚拟化技术配置指导”中的“冗余组”。

1.1.3 以太网冗余子接口

以太网冗余接口是一种三层逻辑接口，它只能处理三层报文。

以太网冗余接口下创建的子接口称为以太网冗余子接口，它也是一种三层虚拟接口，可以配置 IP 地址。主要用来实现在以太网冗余接口上收、发带 VLAN Tag 的二层报文。用户可以在一个以太网冗余接口上配置多个子接口，这样，来自不同 VLAN 的报文可以从不同的子接口进行转发，增强了

设备的组网灵活性，提高了接口利用率。关于以太网冗余接口及其子接口上支持收、发 VLAN Tag 报文的详细描述请参见“二层技术-以太网交换配置指导”中的“VLAN 终结”。

1.2 配置以太网冗余接口

1.2.1 配置限制和指导

1. 支持的成员口类型

以太网冗余接口成员接口的类型可以为：

- 三层以太网接口

  注意

  如果设备面板上标注了接口为 Bypass 接口，或者接口下配置了 Bypass 功能，请不要将这样的三层以太网接口配置为以太网冗余接口的成员接口，以免引起通信异常。Bypass 功能的详细介绍请参见“二层技术-以太网交换配置指导”中的“二层转发”。

- 三层聚合接口
- 上述接口的子接口

2. 成员接口添加限制和指导

一个以太网冗余接口最多可添加两个成员接口，且两个成员接口的优先级不能相同。

只有以太网冗余接口下可以添加成员接口，以太网冗余子接口下不能添加成员接口。

如果以太网冗余接口下创建了子接口，则该以太网冗余接口下的成员接口不能为子接口或者带有子接口的主接口。

一个接口/子接口加入一个以太网冗余接口后，不能加入其它以太网冗余接口。

同一以太网冗余接口的成员接口的类型和速率必须相同，例如均为子接口或者均为 1000M 三层以太网接口，以便保证成员接口切换后不会因带宽不同影响流量转发。

两个成员接口如果都是子接口，则不能是同一主接口的两个子接口，并且其 VLAN 终结配置必须一致。关于 VLAN 终结的详细介绍请参见“二层技术-以太网交换配置指导”中的“VLAN 终结”。

当以太网冗余接口的成员接口包含子接口时，不能指定该以太网冗余接口为 IPv6 静态邻居表项的出接口。关于 IPv6 静态邻居表项的详细描述请参见“三层技术-IP 业务配置指导”中的“IPv6 基础”。

加入以太网冗余接口后，成员接口视图下的配置暂时失效。

3. 成员接口删除限制和指导

当以太网冗余接口中存在成员接口时，请将成员接口从以太网冗余接口中删除后，再删除以太网冗余接口。

1.2.2 配置以太网冗余接口的基本参数

(1) 进入系统视图。

```
system-view
```

(2) 创建以太网冗余接口，并进入该接口视图。

interface reth *interface-number*

(3) 添加成员接口。

member interface *interface-type interface-number* **priority** *priority*

缺省情况下，以太网冗余接口下不存在成员接口。

(4) （可选）配置以太网冗余接口的期望带宽。

bandwidth *bandwidth-value*

缺省情况下，接口的期望带宽为 10000kbps。

期望带宽供业务模块使用，不会对接口实际带宽造成影响。

(5) （可选）配置以太网冗余接口的描述信息。

description *text*

缺省情况下，接口的描述信息为“*接口名* Interface”，比如：Reth1 Interface。

(6) （可选）配置以太网冗余接口的 MTU（Maximum Transmission Unit，最大传输单元）值。

mtu *size*

缺省情况下，以太网冗余接口的 MTU 值为 1500 字节。

(7) 打开以太网冗余接口。

undo shutdown

缺省情况下，以太网冗余接口处于开启状态。

1.2.3 配置激活状态通告参数

1. 功能简介

使用本功能，当以太网冗余接口的成员接口的激活状态切换时，以太网冗余接口会立即发送一次通告报文（免费 ARP、NA 报文），并在接下来的时间，会按照指定时间间隔重复发送指定次数的通告报文，来通告邻居设备成员接口的激活状态发生了变化。

2. 配置限制和指导

如果以太网冗余接口下创建了子接口，以太网冗余接口发送通告报文时，子接口也会发送通告报文，为避免子接口太多，导致通告报文太多，长时间占用 CPU 资源，本命令仅对以太网冗余接口发送的通告报文生效，对子接口发送的通告报文不生效。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 配置当以太网冗余接口的成员接口的激活状态切换时，向邻居设备重新发送通告报文的次数和时间间隔。

reth advertise retransmit *times* **interval** *seconds*

缺省情况下，当以太网冗余接口的成员接口的激活状态切换时，向邻居设备重新发送通告报文的次数为 5，时间间隔为 1 秒。

1.2.4 开启以太网冗余接口的流量快速切换功能

1. 功能简介

缺省情况下，系统能满足大部分场景下以太网冗余接口的流量快速切换需求。只有主设备断电、异常重启情况下，仍要求以太网冗余接口的流量快速切换时，才需要开启以太网冗余接口快速切换功能。

开启本功能后，为了实现以太网冗余接口流量的快速切换，设备允许处于 inactive 状态的以太网冗余接口的成员接口转发报文。这种处理，会小概率导致邻居设备学习到的 MAC 地址表项的出接口为连接 inactive 成员接口的接口，从而使得流量通过冗余组的备节点转发。所以，一般情况下，不建议开启以太网冗余接口快速切换功能。如果需要开启本功能，可以通过配置 **arp timer aging** *aging-time* 命令缩短动态 ARP 表项的老化时间来减少小概率事件的发生。

2. 配置限制和指导

以太网冗余接口中优先级高的成员接口必须位于冗余组的主节点上，并且以太网冗余接口的 active 状态跟随冗余组倒换。

以太网冗余接口下的成员接口必须是物理接口（即不能为聚合接口），该命令才会生效。

用于上行的以太网冗余接口和用于下行的以太网冗余接口都需要配置这个命令。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入以太网冗余接口视图。

interface reth *interface-number*

(3) 开启以太网冗余接口的流量快速切换功能。

fast-switch enable

缺省情况下，以太网冗余接口的流量快速切换功能处于关闭状态。

1.3 配置以太网冗余子接口

1. 功能简介

当以太网冗余接口下的成员接口会同时收到带 VLAN Tag 的报文和三层报文时，可创建以太网冗余子接口，并在以太网冗余子接口下配置 VLAN 终结功能。设备将使用以太网冗余子接口来处理二层报文。

2. 配置限制和指导

请先创建以太网冗余接口，才能创建以太网冗余子接口。

如果以太网冗余接口的成员接口为三层以太网子接口、三层聚合接口子接口，或者成员接口下创建了子接口，则不允许以太网冗余接口下再创建子接口。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 创建以太网冗余子接口，并进入该接口视图。

interface reth *interface-number.subnumber*

如果以太网冗余子接口已创建，执行该命令，则直接进入该以太网冗余子接口的视图。

(3) （可选）配置以太网冗余子接口的期望带宽。

bandwidth *bandwidth-value*

缺省情况下，接口的期望带宽为 10000kbps。

(4) （可选）配置以太网冗余子接口的描述信息。

description *text*

缺省情况下，接口的描述信息为“*接口名* Interface”，比如：Reth1 Interface。

(5) （可选）配置以太网冗余子接口的 MTU 值。

mtu *size*

缺省情况下，以太网冗余子接口的 MTU 值为 1500 字节。

(6) 退回系统视图。

quit

(7) 进入以太网冗余子接口所属的以太网冗余接口视图。

interface reth *interface-number*

(8) 开启以太网冗余子接口的速率统计功能

sub-interface rate-statistic

缺省情况下，以太网冗余子接口的速率统计功能处于关闭状态

配置该命令后，设备会定时统计该接口下所有子接口的速率，用户可以通过 **display interface reth** 命令的 Last 300 seconds input rate 和 Last 300 seconds output rate 字段查看统计结果

(9) 退回系统视图。

quit

(10) 进入以太网冗余子接口视图。

interface reth *interface-number.subnumber*

(11) 打开以太网冗余子接口。

undo shutdown

缺省情况下，以太网冗余子接口处于开启状态。

1.3.2 恢复当前以太网冗余接口/以太网冗余子接口的缺省配置

1. 配置限制和指导

注意

接口下的某些配置恢复到缺省情况后，会对设备上当前运行的业务产生影响。建议您在执行本配置前，完全了解其对网络产生的影响。

您可以在执行 **default** 命令后通过 **display this** 命令确认执行效果。对于未能成功恢复缺省的配置，建议您查阅相关功能的命令手册，手工执行恢复该配置缺省情况的命令。如果操作仍然不能成功，您可以通过设备的提示信息定位原因。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入以太网冗余接口/以太网冗余子接口视图。

interface reth { *interface-number* | *interface-number.subnumber* }

(3) 恢复接口的缺省配置。

default

1.4 以太网冗余接口显示和维护

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后以太网冗余接口的运行情况，通过查看显示信息验证配置的效果。

在用户视图下执行 **reset** 命令可以清除以太网冗余接口的统计信息。

表1-1 以太网冗余显示和维护

操作	命令
显示最近一个统计周期内处于up状态的接口的报文速率统计信息	**display counters rate** { **inbound** | **outbound** } **interface** [**reth** [*interface-number*]]
显示以太网冗余接口/以太网冗余子接口的相关信息	**display interface** [**reth** [*interface-number* | *interface-number.subnumber*]] [**brief** [**description** | **down**]]
显示以太网冗余接口的成员接口的信息	**display reth interface** *interface-type interface-number*
清除以太网冗余接口的统计信息	**reset counters interface** [**reth** [*interface-number*]]

2 冗余组

2.1 冗余组简介

冗余组功能仅在 IRF 模式下支持。在 IRF 组网环境中，冗余组用来实现业务报文的接收、处理、发送都在同一台成员设备上进行。

2.1.1 冗余组工作原理

一个冗余组包含：

- 冗余组节点。一个冗余组必须且最多包含两个冗余组节点。一个为主节点，一个为备节点。每个冗余组节点和一台 IRF 成员设备绑定。
- 冗余组成员：包括物理以太网接口、以太网冗余接口和备份组。冗余组成员部署在和冗余组节点绑定的 IRF 成员设备上。

冗余组节点的主、备状态决定冗余组成员的工作状态。正常情况下，位于主节点上的冗余组成员处于工作状态，位于备节点上的冗余组成员处于冗余备份状态。当主节点故障，主节点变为备节点，冗余组会同时禁用原主节点上的成员，让位于新主节点上的成员工作。从而确保业务报文的接收、处理、发送都在同一台物理设备上进行，两台设备形成设备级备份。如图 2-1 所示，冗余组进行流量切换的步骤大致如下：

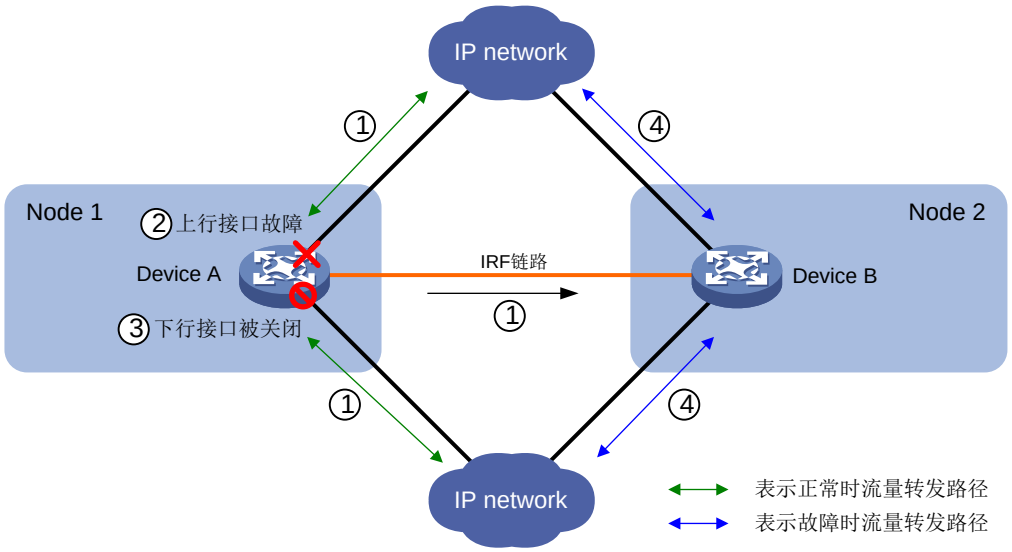
(1) 正常情况下，流量通过 Device A 转发，Device A 上 NAT 等业务的表项和数据备份到 Device B。

(2) Device A 的上行接口故障。

(3) 冗余组关闭 Device A 的下行接口。

(4) 流量迁移到 Device B，通过 Device B 转发。

图2-1 冗余组工作原理示意图



2.1.2 冗余组节点的状态

冗余组节点有两个状态：主和备。和主节点绑定的 IRF 成员设备处理业务、转发报文。

冗余组节点的主备状态由以下因素决定：

- 当冗余组节点绑定的 IRF 成员设备均能正常工作时：
  - 优先级高的为主节点。优先级可通过命令行配置。
  - 当两个节点的优先级相等时，节点编号小的为主节点。节点编号可通过命令行配置。
- 当冗余组节点只绑定了一个 IRF 成员设备或者绑定的两个 IRF 成员设备中有一个不能正常工作时，则与能正常工作的 IRF 成员设备绑定的节点成为主节点。节点能否正常工作由监控机制决定，监控机制的详细介绍请参见“2.1.4 3. 自动倒换/倒回机制”。

2.1.3 冗余组成员

1. 简介

冗余组成员包括物理以太网接口、以太网冗余接口和备份组。其中：

- 物理以太网接口和以太网冗余接口均用于实现流量迁移。两者适用的组网环境不同，根据实际环境选择使用一种即可，具体差异如表 2-1 所示。

表2-1 物理以太网接口和以太网冗余接口差异描述表

接口类型	使用场景	其它说明
物理以太网接口	适用于上行和下行设备运行动态路由协议的场景	加入冗余组的物理以太网接口的类型可以是二层以太网接口、三层以太网接口
以太网冗余接口	适用于上行和下行设备没有运行动态路由协议的场景	以太网冗余接口的成员接口的类型可以是三层以太网接口、三层聚合接口及上述接口的子接口

- 备份组用于实现两台设备间的业务备份。备份组下可以绑定两个 CPU，这两个 CPU 上的数据进行备份。关于备份组的详细介绍请参见“虚拟化技术配置指导”中的“备份组”。

2. 物理以太网接口

冗余组下可以配置节点，节点下可以绑定物理以太网接口。要使冗余组的流量正常切换，需要绑定两组物理以太网接口：

- 一组以太网接口和冗余组的主节点绑定，这组接口必须位于主节点上。该组接口至少需要包含两个物理以太网接口，分别用于上行和下行。
- 一组以太网接口和冗余组的备节点绑定，这组接口必须位于备节点上。该组接口至少需要包含两个物理以太网接口，分别用于上行和下行。

正常情况下，只有位于主节点上的这组接口转发报文，备节点上的这组接口作为主节点上接口组的备份。这两组接口的工作状态由冗余组节点的主备状态决定。如图 2-2 所示，正常情况下，流量通过主节点上的物理以太网接口（Interface A1 和 Interface A2）转发，不通过备节点上的物理以太网接口（Interface B1 和 Interface B2）转发。当主节点上的物理以太网接口（Interface A1）故障，备节点会立即切换成主节点接替原主节点工作，冗余组会关闭原主节点上的其它成员接口，使用新主节点上的成员接口转发报文，如图 2-3 所示。

图2-2 正常情况下，冗余组节点的成员接口工作原理示意图

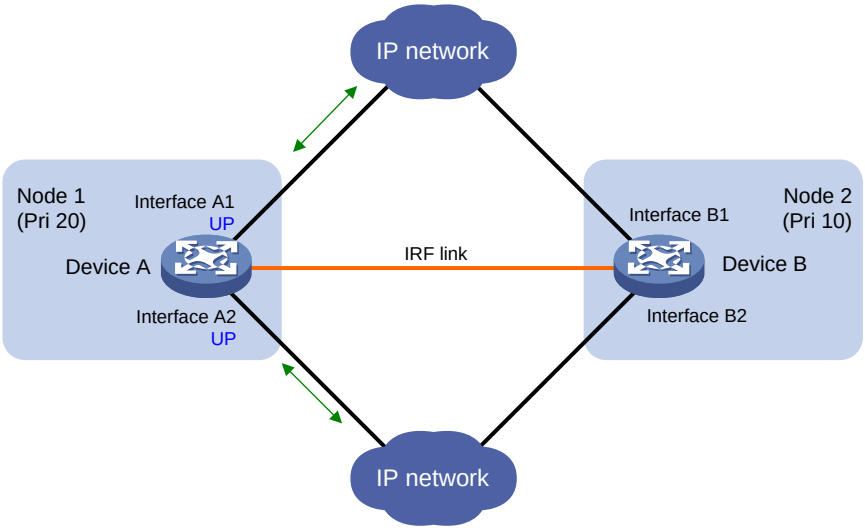
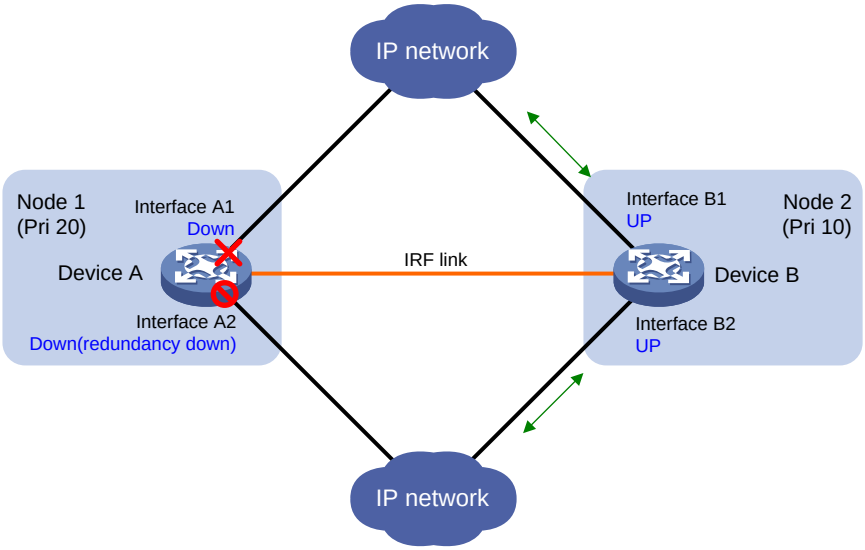


图2-3 上行接口故障时，冗余组节点的成员接口工作原理示意图



3. 以太网冗余接口

冗余组下可以配置以太网冗余接口，以太网冗余接口拥有两个成员接口，这两个成员接口分别位于两个冗余组节点上。一个冗余组需要绑定两个以太网冗余接口，一个用于上行数据迁移，一个用于下行数据迁移。

以太网冗余接口和冗余组节点的联动原理为：冗余组中的主节点正常工作时，以太网冗余接口下优先级高的成员接口处于激活状态。冗余组发生倒换，备节点切换成主节点接替原主节点工作时，以太网冗余接口也发生倒换，让以太网冗余接口下优先级高的成员接口处于非激活状态。

如图 2-4 所示，正常情况下，只有主节点上的以太网冗余接口的成员接口转发报文，备节点上以太网冗余接口的成员接口被冗余组模块关闭。当主节点上以太网冗余接口的成员接口故障，备节点会

立即切换成主节点接替原主节点工作，冗余组会关闭原主节点上其它以太网冗余接口的成员接口，使用新主节点上所有以太网冗余接口的成员接口转发报文，如图 2-5 所示。

图2-4 正常情况下，冗余组和冗余接口联动原理示意图

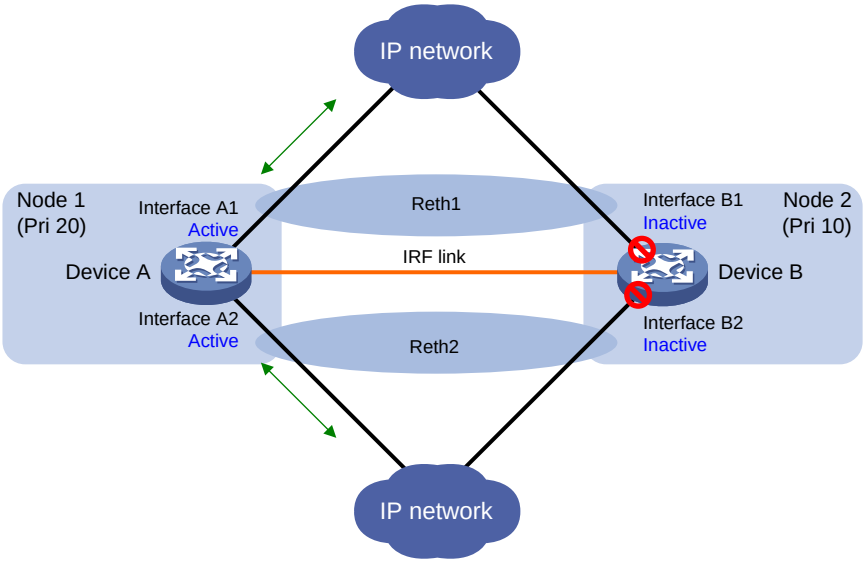
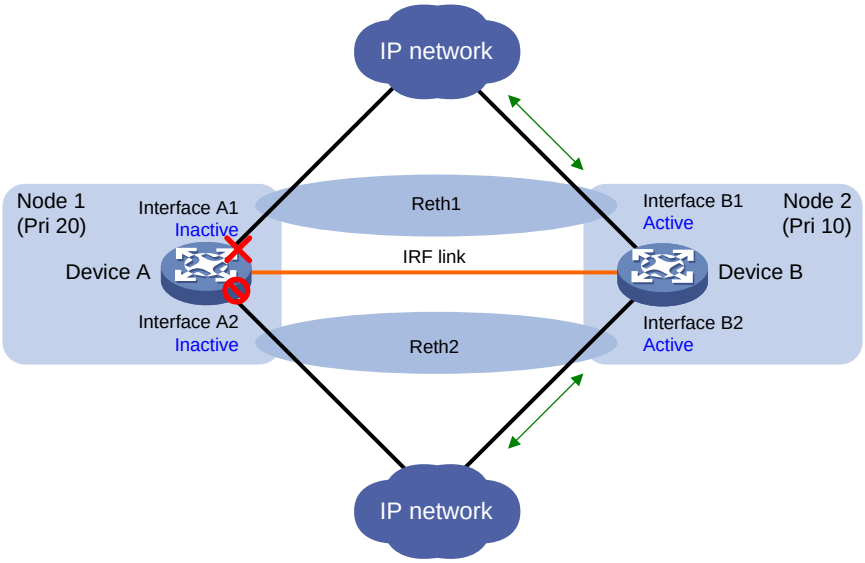


图2-5 上行口故障时，冗余组和冗余接口联动原理示意图



4. 备份组

冗余组下可以配置备份组作为成员，备份组绑定了主、备两个 CPU 成员，其中一个处于激活状态（处理业务报文），另一个处于非激活状态（不工作）。当用户开启业务模块的业务备份功能后，设备会将激活状态的 CPU 成员上的业务数据备份到非激活状态的 CPU 成员上，从而协助实现这两个 CPU 上业务（比如 NAT）的备份。

备份组和冗余组节点的联动原理为：冗余组中的主节点正常工作时，备份组里的主 CPU 成员处于激活状态。冗余组发生倒换，备节点切换成主节点接替原主节点工作时，备份组也发生倒换，让备

份组的备 CPU 成员处于激活状态。冗余组还能通过流量引导来保证倒换前主 CPU 成员处理的流量，倒换后会交给同一备份组中的备 CPU 成员处理，从而保证倒换前后，业务处理的连续性。

2.1.4 冗余组的倒换/倒回机制

1. 功能简介

冗余组的倒换是指系统检测到冗余组的主节点故障，备节点会立即切换成主节点，接替原主节点工作。通过和物理以太网接口、以太网冗余接口以及备份组联动，系统会将流量和业务迁移到新的主节点上处理。

冗余组的倒回是指原主节点故障恢复，系统将流量和业务迁移到原主节点上处理。

根据触发条件不同，冗余组的倒换/倒回机制不同，分为两种：

- 自动倒换/倒回：和 Track 联动来触发倒换和倒回。
- 手工倒换/倒回：由命令行触发倒换和倒回。

2. 自动倒换/倒回定时器

- 保持定时器

  当网络不稳定，监测接口/链路状态频繁改变，会导致 Track 项状态在短时间内频繁改变，从而导致冗余组不断地响应主备倒换事件。使用保持定时器可以避免这种情况的发生。当节点完成主备倒换后，系统启动保持定时器。在保持时间内，不允许再次发生主备倒换。

- 倒回定时器

  当冗余组内优先级高的节点倒回条件就绪时（譬如故障恢复），会触发倒回事件，并启动倒回定时器。由于需要整体倒回，在冗余组倒回的过程中会同时触发很多事件（比如接口状态变化等），这些事件的处理需要时间。倒回定时器能够为冗余组提供一段时间，让节点准备完毕后，再将业务从优先级低的节点倒换到优先级高的节点。

3. 自动倒换/倒回机制

冗余组通过和 Track 联动来实现自动倒换和倒回。

每个冗余组节点都有权重，缺省值为 255，每个冗余组节点必须关联至少一个 Track 项，每个 Track 项对应一个权重增量。当 Track 项变为 NotReady 或 Negative 状态时，冗余组节点用当前权重减去对应的权重增量获得新的当前权重。当 Track 项变为 Positive 时，冗余组节点用当前权重加上对应的权重增量获得新的当前权重。当前权重小于或等于 0 时，则认为该节点故障，无法正常工作，触发冗余组的倒换/倒回。

- 如果是将业务从优先级高的节点倒换到优先级低的节点，则系统收到倒换请求后，等到保持定时器超时后，进行主备倒换。
- 如果是优先级高的节点故障恢复，需要将业务从优先级低的节点倒回，则系统收到整体倒回请求后，等到保持定时器超时后，认为倒回条件就绪，并等到倒回定时器超时后，再进行倒回。

若 Track 模块尚未启动，则节点绑定的 Track 项状态始终为 Positive。关于 Track 的详细介绍请参见“可靠性配置指导”中的“Track”。

4. 手工倒换/倒回机制

如果两个节点均能正常工作，但用户需要更换主节点上的硬件，此时，可手工触发倒换，让业务迁移到优先级低的节点。

当两个节点均能正常工作，但用户未配置 Track 项关联接口时，则系统不能自动倒回，可手工触发倒回，让业务迁移到优先级高的节点。

如果两个节点均能正常工作，但用户将倒回定时器配置为 0，则不允许自动倒回，但可以手工倒回。

2.2 冗余组配置任务简介

冗余组的配置任务如下：

(1) 创建冗余组

(2) 配置冗余组节点

(3) 将接口加入冗余组，以实现流量的迁移。请选择其中一项进行配置。

- 将物理以太网接口加入冗余组

  适用于上行和下行设备运行动态路由协议的场景。

- 将以太网冗余接口加入冗余组

  适用于上行和下行设备没有运行动态路由协议的场景。

(4) 将备份组加入冗余组

(5) 配置冗余组定时器

(6) （可选）手工触发冗余组倒换

(7) （可选）手工触发冗余组倒回

(8) （可选）开启冗余组告警功能

2.3 创建冗余组

1. 配置限制和指导

当冗余组中还存在以太网冗余接口、冗余组节点或者备份组时，不能删除该冗余组。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 创建冗余组，并进入该冗余组视图。

redundancy group *group-name*

2.4 配置冗余组节点

1. 配置限制和指导

一个冗余组下最多可创建两个冗余组节点。不同冗余组下冗余组节点的编号可以相同。

冗余组节点必须和 IRF 成员设备绑定，一个冗余组节点只能和一个 IRF 成员设备绑定。当冗余组节点上存在成员接口或 Track 项时，用户不能取消冗余组节点和 IRF 成员设备的绑定。

关联 Track 项时，需要注意：

- 当 Track 项监控的接口为以太网冗余接口的成员接口或是冗余组节点的成员接口时，请将监控接口配置为关联接口。

- 同一个 Track 项不能与同一冗余组下的两个冗余组节点都关联。当已将某物理接口配置为某冗余组内高优先级冗余组节点的成员接口，或者为某冗余组内以太网冗余接口的高优先级成员接口时，请不要将该物理接口的子接口配置为该冗余组内高优先级冗余组节点的 Track 项关联接口。因为物理接口被协议关闭时，会导致其子接口状态为 Down，该子接口将无法触发自动倒回，此时，需要手工倒回。
- 如果冗余组节点关联的 Track 项监控的是 Blade 接口，则该 Track 项必须使用 **track interface physical** 命令监控该 Blade 接口的物理层状态，否则，将导致冗余组不能正常倒换。

2. 配置准备

请先创建 Track 项，再将该 Track 项和冗余组关联。否则，可能会导致冗余组没有有效的 Track 项而无法触发倒换。关于 Track 项的配置，请参见“可靠性配置指导”中的“Track”。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入冗余组视图。

redundancy group *group-name*

(3) 创建冗余组节点，并进入冗余组节点视图。

node *node-id*

(4) 配置冗余组节点的优先级。

priority *priority*

缺省情况下，冗余组节点的优先级为 1。

(5) 将冗余组节点和 IRF 成员设备绑定。

bind chassis *chassis-number*

缺省情况下，冗余组节点未绑定 IRF 成员设备。

(6) 关联 Track 项。

track *track-entry-number* [**reduced** *weight-reduced*] [**interface** *interface-type interface-number*]

缺省情况下，冗余组节点未关联 Track 项。

2.5 将物理以太网接口加入冗余组

1. 配置限制和指导

冗余组的主节点下至少需要添加两个物理以太网接口，分别用于上行和下行；冗余组的备节点下至少需要添加两个物理以太网接口，分别用于上行和下行。

物理以太网接口只能和一个冗余组节点绑定，不能同时和其它冗余组节点绑定。

本配置中指定的物理以太网接口不能是以太网冗余接口的成员接口。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入冗余组视图。

redundancy group *group-name*

(3) 进入冗余组节点视图。

node *node-id*

(4) 将物理以太网接口和冗余组节点绑定。

node-member interface *interface-type interface-number*

缺省情况下，冗余组节点未绑定物理以太网接口。

2.6 将以太网冗余接口加入冗余组

1. 配置限制和指导

一个冗余组需要绑定两个以太网冗余接口，一个用于上行，一个用于下行。

每个以太网冗余接口必须拥有两个成员接口，这两个成员接口分别位于两个冗余组节点上。

请将位于高优先级冗余组节点上的成员接口的优先级参数配置为更大的值。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 创建以太网冗余接口，并进入该接口视图。

interface reth *interface-number*

(3) 给以太网冗余接口添加成员接口。

member interface *interface-type interface-number* **priority** *priority*

缺省情况下，以太网冗余接口下不存在成员接口。

priority 数值越大，优先级越高。

(4) 退回系统视图。

quit

(5) 进入冗余组视图。

redundancy group *group-name*

(6) 将以太网冗余接口加入冗余组。

member interface reth *interface-number*

缺省情况下，冗余组下不存在以太网冗余接口。

2.7 将备份组加入冗余组

1. 配置限制和指导

备份组的两个成员分别处于冗余组节点绑定的两个 IRF 成员设备上。

请将位于高优先级冗余组节点上的备份组成员属性配置为 Primary。

2. 配置准备

请先创建备份组，否则，无法将备份组加入冗余组。关于备份组的配置请参见“虚拟化技术配置指导”中的“备份组”。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入冗余组视图。

redundancy group *group-name*

(3) 将备份组加入冗余组。

member failover group *group-name*

缺省情况下，冗余组下不存在备份组。

2.8 配置冗余组定时器

(1) 进入系统视图。

system-view

(2) 进入冗余组视图。

redundancy group *group-name*

(3) 配置冗余组节点状态的保持时间，这段时间内不能发生主备倒换。

hold-down-interval *second*

缺省情况下，冗余组节点状态的保持时间为 1 秒。

(4) 配置冗余组节点的倒回延时。

preempt-delay *min*

缺省情况下，冗余组节点的倒回延时为 1 分钟（60 秒）。

如果将倒回时间配置为 0，则表示不允许自动倒回，但可以手工倒回。

2.9 手工触发冗余组倒换

(1) 进入系统视图。

system-view

(2) 进入冗余组视图。

redundancy group *group-name*

(3) 手工触发冗余组进行主备倒换，让冗余组工作在优先级低的节点。

switchover request

2.10 手工触发冗余组倒回

(1) 进入系统视图。

system-view

(2) 进入冗余组视图。

redundancy group *group-name*

(3) 手工触发一次冗余组倒回，让冗余组工作在优先级高的节点。

switchover reset

2.11 开启冗余组告警功能

1. 功能简介

开启冗余组告警功能后，在冗余组人工倒换、故障接口恢复、故障接口生成时，会生成告警信息，并将该信息发送到设备的 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特性。有关告警信息的详细描述，请参见“网络管理和监控配置指导”中的“SNMP”。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 开启冗余组告警功能。

snmp-agent trap enable rddc

缺省情况下，冗余组告警功能处于开启状态。

2.12 冗余组显示和维护

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后冗余组的运行情况，通过查看显示信息验证配置的效果。

表2-2 冗余组显示和维护

操作	命令
显示冗余组的相关信息	**display redundancy group** [*group-name*]

2.13 冗余组典型配置举例

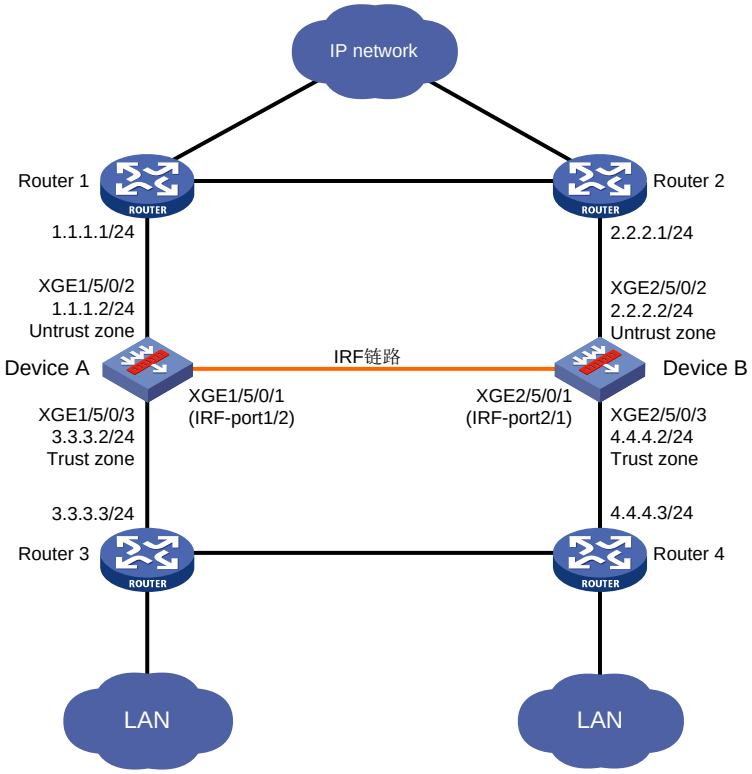
2.13.1 工作在三层，上下行分别连接两台路由器，两台路由器接口不在同一网段

1. 组网需求

- 如图 2-6 所示，Device A 和 Device B 组成 IRF，Router 1 和 IRF 相连的接口与 Router 2 和 IRF 相连的接口不在同一网段，Router 3 和 IRF 相连的接口与 Router 4 和 IRF 相连的接口不在同一网段。
- 正常情况下，流量走 Router 1——Device A——Router 3；当这条通道上的任一链路或者设备故障时，流量切换到 Router 2——Device B——Router 4。正常通道故障恢复时，流量再切回。

2. 组网图

图2-6 配置组网图



3. 配置步骤

(1) 配置 IRF

- 配置 Device A

  # 配置 IRF 端口 1，并将它与物理端口 Ten-GigabitEthernet1/5/0/1 绑定。

```
<DeviceA> system-view
[DeviceA] irf member 1
 Info: Member ID change will take effect after the member reboots and operates in
IRF mode.
[DeviceA] irf-port 2
[DeviceA-irf-port2] port group interface ten-gigabitethernet 1/5/0/1
[DeviceA-irf-port2] quit
```

  # 为确保 Device A 与 Device B 在主设备选举过程中，Device A 为主，修改 Device A 成员优先级为 2（成员优先级大的优先，缺省情况下，设备的成员优先级均为 1）。

```
[DeviceA] irf priority 2
```

  # 将当前配置保存到下次启动配置文件。

```
[DeviceA] quit
<DeviceA> save
```

  # 将设备的运行模式切换到 IRF 模式。

```
<DeviceA> system-view
[DeviceA] chassis convert mode irf
The device will switch to IRF mode and reboot. You are recommended to save the current
running configuration and specify the configuration file for the next startup.
Continue? [Y/N]:y
 Do you want to convert the content of the next startup configuration file
flash:/startup.cfg to make it available in IRF mode? [Y/N]:y
 Please wait...
 Saving the converted configuration file to the main board succeeded.
Slot 1:
 Saving the converted configuration file succeeded.
 Now rebooting, please wait...
```

设备重启后 Device A 组成了只有一台成员设备的 IRF。

- 配置 Device B

配置 Device B 的成员编号为 2，创建 IRF 端口 1，并将它与物理端口 Ten-GigabitEthernet1/5/0/1 绑定。

```
<DeviceB> system-view
[DeviceB] irf member 2
 Info: Member ID change will take effect after the member reboots and operates in
IRF mode.
[DeviceB] irf-port 1
[DeviceB-irf-port1] port group interface ten-gigabitethernet 1/5/0/1
[DeviceB-irf-port1] quit
```

将当前配置保存到下次启动配置文件。

```
[DeviceB] quit
<DeviceB> save
```

参照图 2-6 进行物理连线。

将设备的运行模式切换到 IRF 模式。

```
<DeviceB> system-view
[DeviceB] chassis convert mode irf
The device will switch to IRF mode and reboot. You are recommended to save the current
running configuration and specify the configuration file for the next startup.
Continue? [Y/N]:y
 Do you want to convert the content of the next startup configuration file
flash:/startup.cfg to make it available in IRF mode? [Y/N]:y
 Please wait...
 Saving the converted configuration file to the main board succeeded.
Slot 1:
 Saving the converted configuration file succeeded.
```

```
Now rebooting, please wait...
```

设备 B 重启后与设备 A 形成 IRF。

(2) 配置 Track，监测上、下行接口的状态以及安全引擎上 Blade 接口的物理状态。

```
<DeviceA> system-view
[DeviceA] track 1 interface ten-gigabitethernet 1/5/0/2 physical
[DeviceA-track-1] quit
[DeviceA] track 2 interface ten-gigabitethernet 1/5/0/3 physical
[DeviceA-track-2] quit
[DeviceA] track 3 interface ten-gigabitethernet 2/5/0/2 physical
[DeviceA-track-3] quit
[DeviceA] track 4 interface ten-gigabitethernet 2/5/0/3 physical
[DeviceA-track-4] quit
[DeviceA] track 5 interface blade 1/4/0/1 physical
[DeviceA-track-5] quit
[DeviceA] track 6 interface blade 2/4/0/1 physical
[DeviceA-track-6] quit
```

(3) 配置备份组，并指定 Device A 上的安全引擎为主，Device B 上的安全引擎为备。

```
[DeviceA] failover group group1
[DeviceA-failover-group-group1] bind chassis 1 slot 4 cpu 1 primary
[DeviceA-failover-group-group1] bind chassis 2 slot 4 cpu 1 secondary
[DeviceA-failover-group-group1] quit
```

(4) 配置冗余组。

创建 Node 1，Node 1 和 Device A 绑定，为主节点，成员接口为 Ten-GigabitEthernet1/5/0/2 和 Ten-GigabitEthernet1/5/0/3。关联的 Track 项为 1、2 和 5。

```
[DeviceA] redundancy group aaa
[DeviceA-redundancy-group-aaa] node 1
[DeviceA-redundancy-group-aaa-node1] bind chassis 1
[DeviceA-redundancy-group-aaa-node1] priority 100
[DeviceA-redundancy-group-aaa-node1] node-member interface ten-gigabitethernet
1/5/0/2
[DeviceA-redundancy-group-aaa-node1] node-member interface ten-gigabitethernet
1/5/0/3
[DeviceA-redundancy-group-aaa-node1] track 1 interface ten-gigabitethernet 1/5/0/2
[DeviceA-redundancy-group-aaa-node1] track 2 interface ten-gigabitethernet 1/5/0/3
[DeviceA-redundancy-group-aaa-node1] track 5 interface blade 1/4/0/1
[DeviceA-redundancy-group-aaa-node1] quit
```

创建 Node 2，Node 2 和 Device B 绑定，为备节点，成员接口为 Ten-GigabitEthernet2/5/0/2 和 Ten-GigabitEthernet2/5/0/3。关联的 Track 项为 3、4 和 6。

```
[DeviceA-redundancy-group-aaa] node 2
```

```
[DeviceA-redundancy-group-aaa-node2] bind chassis 2
[DeviceA-redundancy-group-aaa-node2] priority 50
[DeviceA-redundancy-group-aaa-node2] node-member interface ten-gigabitethernet 2/5/0/2
[DeviceA-redundancy-group-aaa-node2] node-member interface ten-gigabitethernet 2/5/0/3
[DeviceA-redundancy-group-aaa-node2] track 3 interface ten-gigabitethernet 2/5/0/2
[DeviceA-redundancy-group-aaa-node2] track 4 interface ten-gigabitethernet 2/5/0/3
[DeviceA-redundancy-group-aaa-node2] track 6 interface blade 2/4/0/1
[DeviceA-redundancy-group-aaa-node2] quit
```

将备份组 1 添加到冗余组中。

```
[DeviceA-redundancy-group-aaa] member failover group group1
[DeviceA-redundancy-group-aaa] quit
```

(5) 配置接口 IP 地址

根据组网图中规划的信息，配置各接口的 IP 地址，具体配置步骤如下。

```
[DeviceA] interface ten-gigabitethernet 1/5/0/2
[DeviceA-Ten-GigabitEthernet1/5/0/2] ip address 1.1.1.2 255.255.255.0
[DeviceA-Ten-GigabitEthernet1/5/0/2] quit
```

请参考以上步骤配置其他接口的 IP 地址，具体配置步骤略。

(6) 配置静态路由

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由。本举例假设 LAN 网段为 5.5.5.0/24，实际使用中请以具体组网情况为准，具体配置步骤如下。

```
[DeviceA] ip route-static 0.0.0.0 0 1.1.1.1
[DeviceA] ip route-static 0.0.0.0 0 2.2.2.1 preference 80
[DeviceA] ip route-static 5.5.5.0 24 3.3.3.3
[DeviceA] ip route-static 5.5.5.0 24 4.4.4.3 preference 80
```

(7) 配置接口加入安全域

请根据组网图中规划的信息，将接口加入对应的安全域，具体配置步骤如下。

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface ten-gigabitethernet 1/5/0/2
[DeviceA-security-zone-Untrust] import interface ten-gigabitethernet 2/5/0/2
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface ten-gigabitethernet 1/5/0/3
[DeviceA-security-zone-Trust] import interface ten-gigabitethernet 2/5/0/3
[DeviceA-security-zone-Trust] quit
```

(8) 配置安全策略

配置名称为 trust-untrust 的安全策略规则，使 LAN 1 和 LAN 2 中的主机可以访问外网，具体配置步骤如下。

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule 1 name trust-untrust
[DeviceA-security-policy-ip-1-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-1-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-1-trust-untrust] source-ip-subnet 5.5.5.0 24
[DeviceA-security-policy-ip-1-trust-untrust] action pass
[DeviceA-security-policy-ip-1-trust-untrust] quit
[DeviceA-security-policy-ip]quit
```

4. 验证配置

(1) 缺省情况下的显示信息

显示冗余组信息。可以看到优先级高的 Node 1 为主节点，Node 1 和 Node 2 下面的成员接口都处于 UP 状态。

```
[DeviceA] display redundancy group aaa
Redundancy group aaa (ID 1):
  Node ID      Chassis       Priority   Status        Track weight
  1            Chassis1      100        Primary       255
  2            Chassis4      50         Secondary     255

Preempt delay time remained     : 0    min
Preempt delay timer setting     : 1    min
Remaining hold-down time        : 0    sec
Hold-down timer setting         : 1    sec
Manual switchover request       : No

Member interfaces:
Member failover groups:
    group1

Node 1:
  Node member     Physical status
      XGE1/5/0/2   UP
      XGE1/5/0/3   UP
  Track info:
    Track    Status       Reduced weight     Interface
    1        Positive     255                XGE1/5/0/2
```

```
    2        Positive     255               XGE1/5/0/3
    5        Positive     255               Blade1/4/0/1
Node 2:
  Node member     Physical status
      XGE2/5/0/2   UP
      XGE2/5/0/3   UP
  Track info:
    Track    Status       Reduced weight     Interface
    3        Positive       255              XGE2/5/0/2
    4        Positive       255              XGE2/5/0/3
    6        Positive       255              Blade2/4/0/1
```

显示备份组信息。可以看到备份组中配置为 Primary 的安全引擎处理业务。

```
[DeviceA] display failover group group1
Stateful failover group information:
ID   Name                      Primary         Secondary       Active Status
255  group1                    1/4.1           2/4.1           Primary
```

(2) 冗余组内主备倒换后的显示信息

手工关闭接口 Ten-GigabitEthernet1/5/0/3，显示冗余组信息。可以看到优先级低的 Node 2 为主节点，Node 1 的成员接口 Ten-GigabitEthernet1/5/0/3 故障（DOWN），Ten-GigabitEthernet1/5/0/2 被协议关闭（DOWN(redundancy down)），Node 2 的成员接口转发报文。

```
[DeviceA] interface ten-gigabitethernet 1/5/0/3
[DeviceA-Ten-GigabitEthernet1/5/0/3] shutdown
[DeviceA-Ten-GigabitEthernet1/5/0/3] quit
[DeviceA] display redundancy group aaa
Redundancy group aaa (ID 1):
  Node ID      Chassis       Priority   Status        Track weight
  1            Chassis1      100        Secondary     -255
  2            Chassis4      50         Primary       255

Preempt delay time remained     : 0    min
Preempt delay timer setting     : 1    min
Remaining hold-down time        : 0    sec
Hold-down timer setting         : 1    sec
Manual switchover request       : No

Member interfaces:
Member failover groups:
```

```
    group1


Node 1:
  Node member     Physical status
      XGE1/5/0/2        DOWN(redundancy down)
      XGE1/5/0/3        DOWN
  Track info:
    Track    Status       Reduced weight     Interface
    1         Negative     255                XGE1/5/0/2
    2         Negative     255                XGE1/5/0/3 (Fault)
    5         Positive     255                Blade1/4/0/1
Node 2:
  Node member     Physical status
      XGE2/5/0/2    UP
      XGE2/5/0/3    UP
  Track info:
    Track    Status       Reduced weight     Interface
    3         Positive       255              XGE2/5/0/2
    4         Positive       255              XGE2/5/0/3
    6         Positive       255              Blade2/4/0/1
```

显示备份组信息。可以看到备份组中配置为 Secondary 的安全引擎处理业务。

```
[DeviceA] display failover group group1
Stateful failover group information:
ID   Name                      Primary         Secondary       Active Status
255  group1                    1/4.1           2/4.1           Secondary
```

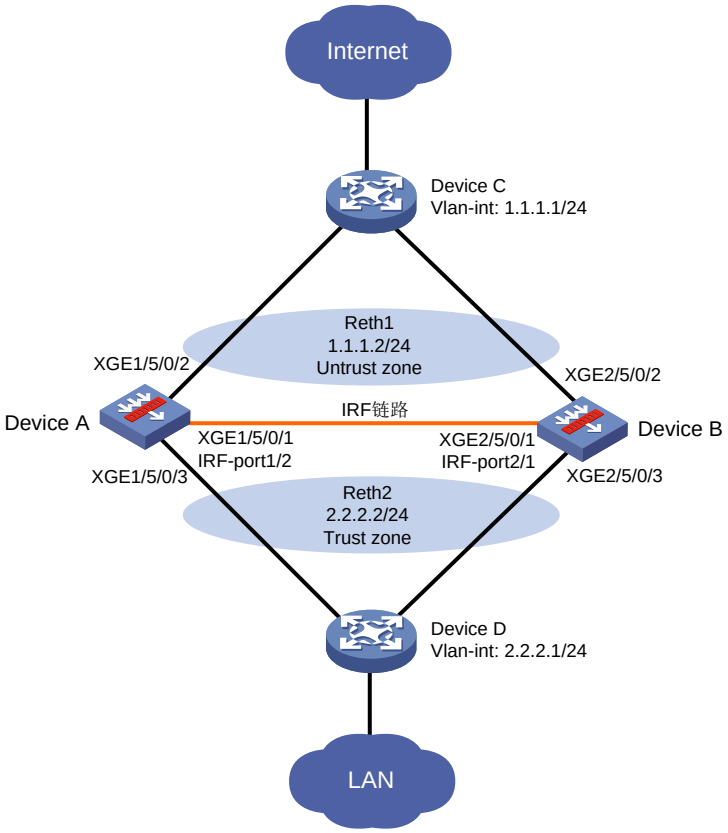
2.13.2 工作在三层，上下行分别连接一台设备

1. 组网需求

- 如图 2-7 所示，Device A 和 Device B 组成 IRF，Device A 和 Device B 分别用一个接口组建冗余接口，连接上行设备 Device C；再分别用一个接口组建冗余接口，连接下行设备 Device D。
- Device C 使用 VLAN 接口连接 Device A 和 Device B，Device D 也使用 VLAN 接口连接 Device A 和 Device B。
- 正常情况下，流量走 Device D——Device A——Device C；当这条通道上的任一链路或者设备故障时，流量切换到 Device D——Device B——Device C。正常通道故障恢复时，流量再切回。

2. 组网图

图2-7 配置组网图



3. 配置准备

请参见“虚拟化配置指导”中的“IRF”，将 Device A 和 Device B 组成 IRF。本文只描述 IRF 组成后，在 IRF 上配置冗余组。

4. 配置步骤

(1) 配置 IRF。配置步骤请参见“2.13.1 ”。

(2) 配置以太网冗余接口

创建 Reth1，IP 地址为 1.1.1.2/24，成员接口为 Ten-GigabitEthernet1/5/0/2 和 Ten-GigabitEthernet2/5/0/2，其中 Ten-GigabitEthernet1/5/0/2 的优先级为 255，Ten-GigabitEthernet2/5/0/2 的优先级为 50。

```
<DeviceA> system-view
[DeviceA] interface reth 1
[DeviceA-Reth1] ip address 1.1.1.2 24
[DeviceA-Reth1] member interface ten-gigabitethernet 1/5/0/2 priority 255
[DeviceA-Reth1] member interface ten-gigabitethernet 2/5/0/2 priority 50
[DeviceA-Reth1] quit
```

创建 Reth2，IP 地址为 2.2.2.2/24，成员接口为 Ten-GigabitEthernet1/5/0/3 和 Ten-GigabitEthernet2/5/0/3，其中 Ten-GigabitEthernet1/5/0/3 的优先级为 255，Ten-GigabitEthernet2/5/0/3 的优先级为 50。

```
[DeviceA] interface reth 2
[DeviceA-Reth2] ip address 2.2.2.2 24
[DeviceA-Reth2] member interface ten-gigabitethernet 1/5/0/3 priority 255
[DeviceA-Reth2] member interface ten-gigabitethernet 2/5/0/3 priority 50
[DeviceA-Reth2] quit
```

(3) 配置 Track，监测上、下行接口的状态以及安全引擎上 Blade 接口的物理状态。

```
[DeviceA] track 1 interface ten-gigabitethernet 1/5/0/2 physical
[DeviceA-track-1] quit
[DeviceA] track 2 interface ten-gigabitethernet 1/5/0/3 physical
[DeviceA-track-2] quit
[DeviceA] track 3 interface ten-gigabitethernet 2/5/0/2 physical
[DeviceA-track-3] quit
[DeviceA] track 4 interface ten-gigabitethernet 2/5/0/3 physical
[DeviceA-track-4] quit
[DeviceA] track 5 interface blade 1/4/0/1 physical
[DeviceA-track-5] quit
[DeviceA] track 6 interface blade 2/4/0/1 physical
[DeviceA-track-6] quit
```

(4) 配置备份组，并指定 Device A 上的安全引擎为主，Device B 上的安全引擎为备。

```
[DeviceA] failover group group1
[DeviceA-failover-group-group1] bind chassis 1 slot 4 cpu 1 primary
[DeviceA-failover-group-group1] bind chassis 2 slot 4 cpu 1 secondary
[DeviceA-failover-group-group1] quit
```

(5) 配置冗余组

创建 Node 1，Node 1 和 Device A 绑定，为主节点。关联的 Track 项为 1、2 和 5。

```
[DeviceA] redundancy group aaa
[DeviceA-redundancy-group-aaa] node 1
[DeviceA-redundancy-group-aaa-node1] bind chassis 1
[DeviceA-redundancy-group-aaa-node1] priority 100
[DeviceA-redundancy-group-aaa-node1] track 1 interface ten-gigabitethernet 1/5/0/2
[DeviceA-redundancy-group-aaa-node1] track 2 interface ten-gigabitethernet 1/5/0/3
[DeviceA-redundancy-group-aaa-node1] track 5 interface blade 1/4/0/1
[DeviceA-redundancy-group-aaa-node1] quit
```

创建 Node 2，Node 2 和 Device B 绑定，为备节点。关联的 Track 项为 3、4 和 6。

```
[DeviceA-redundancy-group-aaa] node 2
```

```
[DeviceA-redundancy-group-aaa-node2] bind chassis 2
[DeviceA-redundancy-group-aaa-node2] priority 50
[DeviceA-redundancy-group-aaa-node2] track 3 interface ten-gigabitethernet 2/5/0/2
[DeviceA-redundancy-group-aaa-node2] track 4 interface ten-gigabitethernet 2/5/0/3
[DeviceA-redundancy-group-aaa-node2] track 6 interface blade 2/4/0/1
[DeviceA-redundancy-group-aaa-node2] quit
```

将 Reth1、Reth2 和备份组 1 添加到冗余组中。

```
[DeviceA-redundancy-group-aaa] member interface reth 1
[DeviceA-redundancy-group-aaa] member interface reth 2
[DeviceA-redundancy-group-aaa] member failover group group1
[DeviceA-redundancy-group-aaa] quit
```

(6) 配置静态路由

本举例仅以静态路由方式配置路由信息。实际组网中，请根据具体情况选择相应的路由配置方式。

请根据组网图中规划的信息，配置静态路由。本举例假设 LAN 的网段为 3.3.3.0/24，实际使用中请以具体组网情况为准，具体配置步骤如下。

```
[DeviceA] ip route-static 0.0.0.0 0 1.1.1.1
[DeviceA] ip route-static 3.3.3.0 24 2.2.2.1
```

(7) 配置接口加入安全域

请根据组网图中规划的信息，将接口加入对应的安全域，具体配置步骤如下。

```
[DeviceA] security-zone name untrust
[DeviceA-security-zone-Untrust] import interface reth 1
[DeviceA-security-zone-Untrust] quit
[DeviceA] security-zone name trust
[DeviceA-security-zone-Trust] import interface reth 2
[DeviceA-security-zone-Trust] quit
```

(8) 配置安全策略

配置名称为 trust-untrust 的安全策略规则，使 LAN 中的主机可以访问外网，具体配置步骤如下。

```
[DeviceA] security-policy ip
[DeviceA-security-policy-ip] rule 1 name trust-untrust
[DeviceA-security-policy-ip-1-trust-untrust] source-zone trust
[DeviceA-security-policy-ip-1-trust-untrust] destination-zone untrust
[DeviceA-security-policy-ip-1-trust-untrust] source-ip-subnet 3.3.3.0 24
[DeviceA-security-policy-ip-1-trust-untrust] action pass
[DeviceA-security-policy-ip-1-trust-untrust] quit
[DeviceA-security-policy-ip] quit
```

5. 验证配置

(1) 缺省情况下的显示信息

显示冗余组信息。可以看到优先级高的 Node 1 为主节点，Reth1 和 Reth2 中优先级高的成员接口处于激活状态。

```
[DeviceA] display redundancy group aaa
Redundancy group aaa (ID 1):
  Node ID   Chassis     Priority  Status       Track weight
  Node1     chassis1    100       Primary      255
  Node2     chassis2    50        Secondary    255

Preempt delay time remained   : 0    min
Preempt delay timer setting   : 1    min
Remaining hold-down time      : 0    sec
Hold-down timer setting       : 300  sec
Manual switchover request     : No

Member interfaces:
    Reth1          Reth2
Member failover groups:
    group1

Node 1:
  Track info:
    Track     Status        Reduced weight       Interface
    1         Positive      255                  XGE1/5/0/2
    2         Positive      255                  XGE1/5/0/3
    5         Positive      255                  Blade1/4/0/1
Node 2:
  Track info:
    Track     Status        Reduced weight      Interface
    3         Positive      255                 XGE2/5/0/2
    4         Positive      255                 XGE2/5/0/3
    6         Positive      255                 Blade2/4/0/1
```

显示 Reth 信息。可以看到 Reth1 和 Reth2 中优先级高的成员接口处于激活状态。

```
[DeviceA] display reth interface reth 1
Reth1 :
  Redundancy group  : aaa
  Member               Physical status        Forwarding status    Presence status
```

```
  XGE1/5/0/2              UP                      Active                Normal
  XGE2/5/0/2              UP                      Inactive              Normal
[DeviceA] display reth interface reth 2
Reth2 :
  Redundancy group  : aaa
  Member                Physical status         Forwarding status    Presence status
  XGE1/5/0/3              UP                      Active                Normal
  XGE2/5/0/3              UP                      Inactive              Normal
```

显示备份组信息。可以看到备份组中配置为 Primary 的安全引擎处理业务。

```
[DeviceA] display failover group group1
Stateful failover group information:
ID   Name                     Primary         Secondary       Active Status
255  group1                   1/4.1           2/4.1           Primary
```

(2) 冗余组内主备倒换后的显示信息

手工关闭接口 Ten-GigabitEthernet1/5/0/3，显示冗余组信息。可以看到优先级低的 Node 2 为主节点。

```
[DeviceA] interface ten-gigabitethernet 1/5/0/3
[DeviceA-Ten-GigabitEthernet1/5/0/3] shutdown
[DeviceA-Ten-GigabitEthernet1/5/0/3] quit
[DeviceA] display redundancy group aaa
Redundancy group aaa (ID 1):
  Node ID   Chassis     Priority  Status       Track weight
  Node1     chassis1    100       Secondary    -255
  Node2     chassis2    50        Primary      255

Preempt delay time remained   : 0    min
Preempt delay timer setting   : 1    min
Remaining hold-down time      : 0    sec
Hold-down timer setting       : 300  sec
Manual switchover request     : No

Member interfaces:
    Reth1          Reth2
Member failover groups:
    group1

Node 1:
  Track info:
```

```
    Track     Status       Reduced weight     Interface
    1         Negative     255                XGE1/5/0/2
    2         Negative     255                XGE1/5/0/3(Fault)
    5         Positive     255                Blade1/4/0/1
Node 2:
  Track info:
    Track     Status       Reduced weight     Interface
    3         Positive     255                XGE2/5/0/2
    4         Positive     255                XGE2/5/0/3
    6         Positive     255                Blade2/4/0/1
```

显示 Reth 的信息。Reth2 下的接口 Ten-GigabitEthernet1/5/0/3 故障（DOWN），Reth1 下的接口 Ten-GigabitEthernet1/5/0/2 被协议关闭（DOWN(redundancy down)）。Ten-GigabitEthernet2/5/0/2 和 Ten-GigabitEthernet2/5/0/3 激活。

```
[DeviceA] display reth interface reth 1
Reth1 :
Redundancy group  : aaa
  Member                Physical status         Forwarding status    Presence status
  XGE1/5/0/2              DOWN(redundancy down)   Inactive           Normal
  XGE2/5/0/2              UP                      Active             Normal
[DeviceA] display reth interface reth 2
Reth2 :
Redundancy group  : aaa
  Member                Physical status         Forwarding status    Presence status
  XGE1/5/0/3              DOWN                    Inactive           Normal
  XGE2/5/0/3              UP                      Active             Normal
```

显示备份组信息。可以看到备份组中配置为 Secondary 的安全引擎处理业务。

```
[DeviceA] display failover group group1
Stateful failover group information:
ID   Name                     Primary         Secondary       Active Status
255  group1                   1/4.1           2/4.1           Secondary
```

目 录

1 备份组 ····· 1-1
1.1 备份组简介 ····· 1-1
1.1.1 备份组工作机制 ····· 1-1
1.1.2 备份组分类 ····· 1-1
1.2 备份组配置限制和指导 ····· 1-1
1.3 配置手动备份组 ····· 1-1
1.4 备份组显示和维护 ····· 1-2

1 备份组

1.1 备份组简介

备份组用于协助特定业务模块（例如（例如负载均衡））实现业务数据在指定 CPU（又称为“节点”）间的备份，为特定业务的可靠运行提供保障。

1.1.1 备份组工作机制

一个备份组由两个节点组成，一个为主节点，一个为备节点。通常情况下，主节点处于激活状态，处理业务报文；备节点处于非激活状态，不处理业务报文，为主节点提供冗余备份。如果主节点故障，则备节点会变成激活状态，接替主节点工作。

业务模块引用备份组，并开启业务备份功能后，业务模块会将业务数据从激活状态的节点备份到非激活状态的节点。哪些模块可以引用备份组，请参见业务模块的配置指导手册。

1.1.2 备份组分类

备份组有两种：自动备份组和手动备份组。

1. 自动备份组

当安全业务板插入设备时，系统自动为每块安全业务板上的安全引擎创建一个备份组，这种备份组称为自动备份组，这个安全引擎即为自动备份组的主节点。

正常情况下自动备份组没有备节点，不进行业务备份。当设备存在多个安全引擎时，通过指定自动备份组，用户可以让业务在指定的安全引擎上运行。当安全引擎故障时，系统会自动将流量平均分配到同一安全引擎组内的其它安全引擎上处理。有关安全引擎和安全引擎组的详细介绍，请参见“虚拟化技术配置指导”中的“Context”。

2. 手动备份组

用户通过命令行创建的备份组。

手动备份组的主节点和备节点可通过命令行配置。

1.2 备份组配置限制和指导

为了保证业务在主、备节点迁移后，仍能正常运行，建议将不同单板上的性能相当的两个 CPU 互为备份。

1.3 配置手动备份组

(1) 进入系统视图。

system-view

(2) 创建备份组，并进入备份组视图。

failover group *group-name*

缺省情况下，存在自动备份组（名称以 AutoBackup 为前缀），不存在手动备份组。

(3) 将节点加入手动备份组。

（独立运行模式）

bind slot *slot-number* **cpu** *cpu-number* { **primary** | **secondary** }

（IRF 模式）

bind chassis *chassis-number* **slot** *slot-number* **cpu** *cpu-number* { **primary** | **secondary** }

缺省情况下，备份组内不存在节点。

不同备份组的主节点不能相同，同一备份组的主节点和备节点不能相同。

1.4 备份组显示和维护

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后备份组的运行情况，通过查看显示信息验证配置的效果。

表1-1 备份组显示和维护

操作	命令
显示备份组的信息	**display failover group** [*group-name*]