H3C SecPath 防火墙产品

虚拟化技术配置指导(V7)

新华三技术有限公司
http://www.h3c.com

资料版本：6W303-20190927

前 言

本配置指导介绍了各软件特性的原理及其配置方法，包含原理简介、配置任务描述和配置举例。《虚拟化技术配置指导》主要介绍 IRF 和 Context 相关的特性。

前言部分包含如下内容：

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

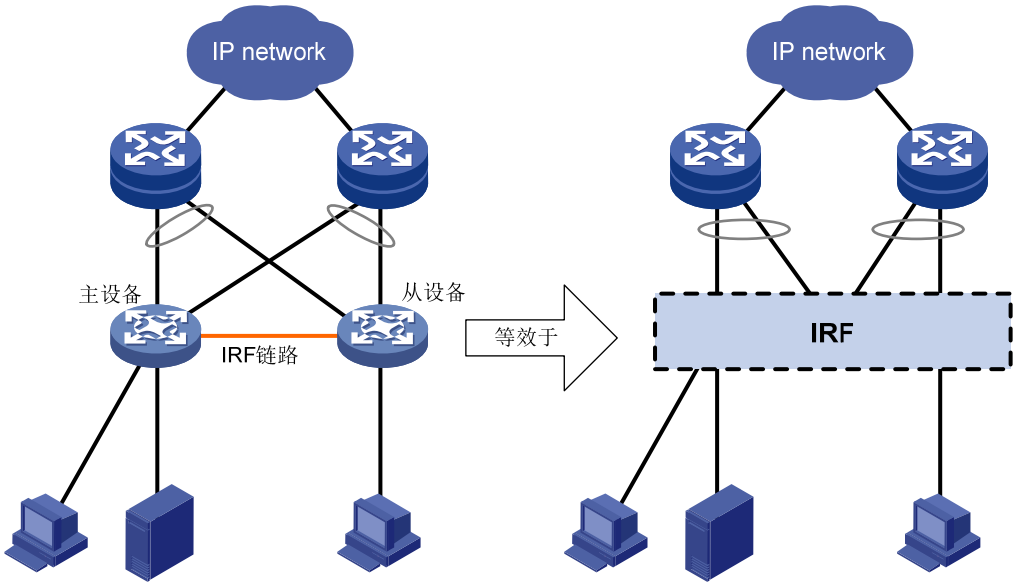
1.1 IRF简介

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



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. 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”。

2. 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。

3. 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。

4. 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。

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。

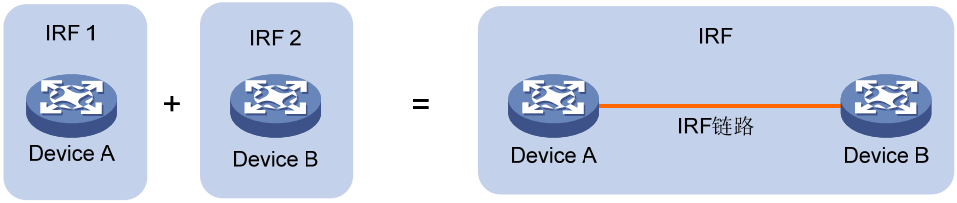
5. IRF物理端口

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。

6. IRF合并

如 图 1-2 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。

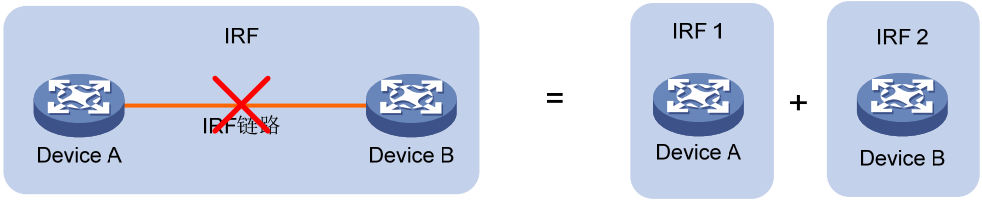
图1-2 IRF 合并示意图



7. IRF分裂

如 图 1-3 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1-3 IRF 分裂示意图



8. MAD

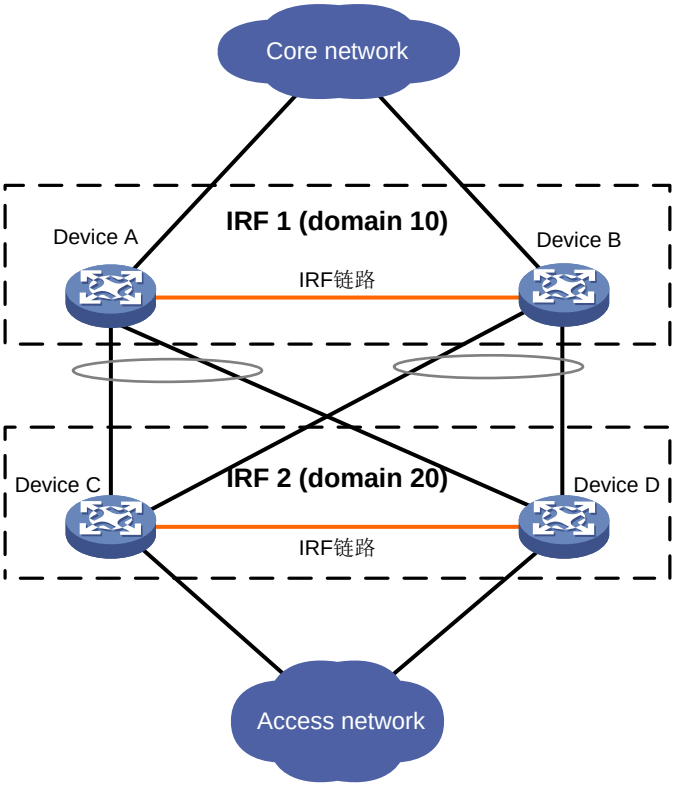
IRF 链路故障会导致一个 IRF 分裂成多个新的 IRF。这些 IRF 拥有相同的 IP 地址等三层配置，会引起地址冲突，导致故障在网络中扩大。MAD（Multi-Active Detection，多 Active 检测）机制用来进行 IRF 分裂检测、冲突处理和故障恢复，从而提高系统的可用性。

9. 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 1-4 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1-4 多 IRF 域示意图

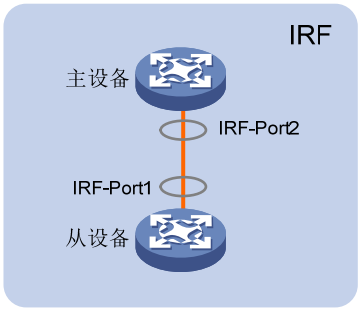


1.1.4 IRF的连接拓扑

IRF的连接拓扑为链形连接，如 图 1-5 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1-5 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

接口编号采用成员设备编号/槽位编号/接口序号的格式，其中：

- 成员设备编号：用来标志不同成员设备上的接口。
- 接口序号：与设备支持的接口数量相关，请查看设备前面板上的丝印。

例如，将成员编号为 2 的从设备上第一个固定端口的链路类型设置为 Trunk，可参照以下步骤：

```
<Sysname> system-view
[Sysname] interface gigabitethernet 2/0/1
[Sysname-GigabitEthernet2/0/1] port link-type trunk
```

1.1.7 IRF中的文件系统命名规则

使用存储介质的名称可以访问主设备的文件系统，使用“slotMember-ID#存储介质的名称”可以访问从设备的文件系统。例如：

- 创建并显示 IRF 中主设备存储介质 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 中从设备（成员编号为 2）存储介质 Flash 根目录下的 test 文件夹：

```
<Master> mkdir slot2#flash:/test
Creating directory slot2#flash:/test... Done.
<Master> cd slot2#flash:/test
<Master> dir
Directory of slot2#flash:/test
The directory is empty.

524288 KB total (128812 KB free)
```

1.1.8 IRF中的配置文件同步

IRF 技术使用了严格的配置文件同步机制，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 中的从设备在启动时，会自动寻找主设备，并将主设备的当前配置文件同步到本地并执行；如果 IRF 中的所有设备同时启动，则从设备会将主设备的起始配置文件同步至本地并执行。
- 在 IRF 正常工作后，用户所进行的任何配置，都会记录到主设备的当前配置文件中，并同步到 IRF 中的各个设备执行。

通过即时的同步，IRF 中所有设备均保存相同的配置文件，即使主设备出现故障，其它设备仍能够按照相同的配置文件执行各项功能。

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）或者BFD（Bidirectional Forwarding Detection，双向转发检测）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“可靠性配置指导”中的“BFD”。

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。

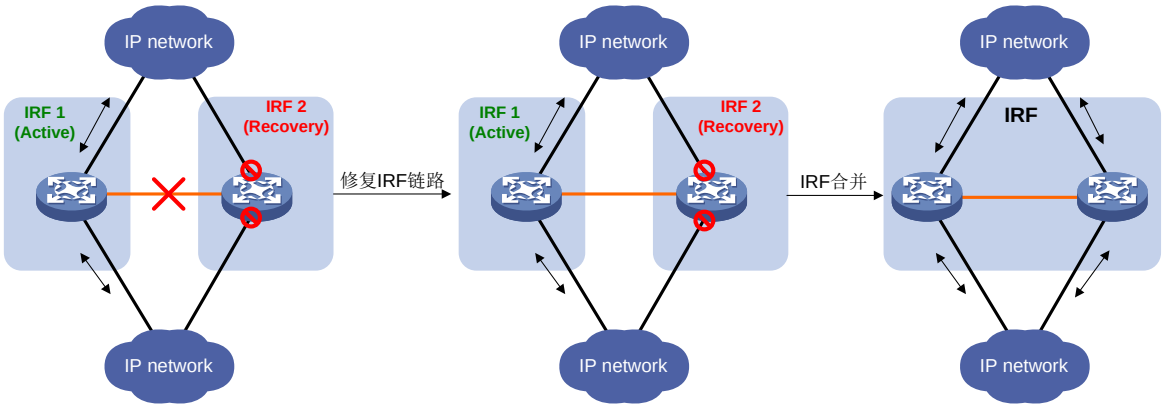
IRF 链路修复后，系统会给出提示信息要求用户手工重启处于 Recovery 状态的 IRF。

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 1-6 所示。

注意

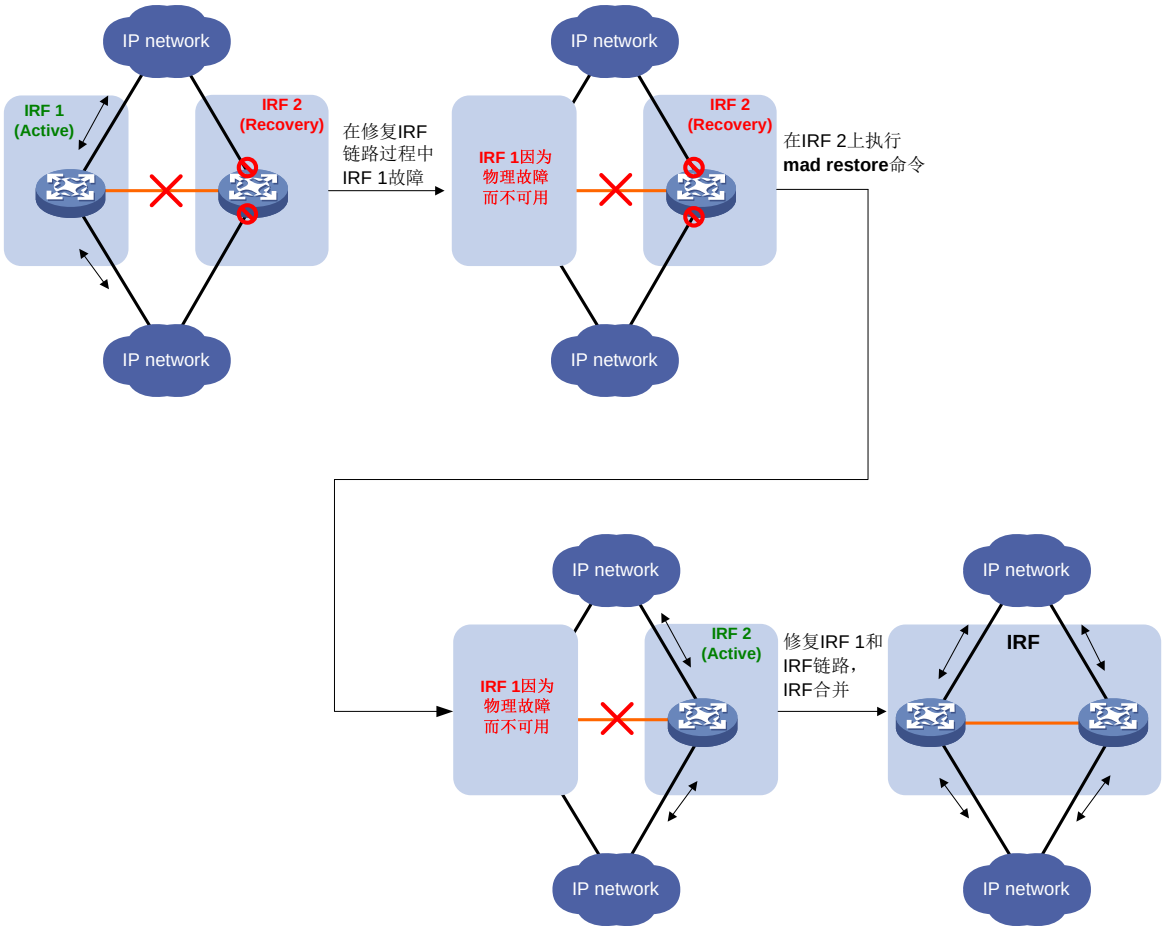
- 请根据提示重启处于Recovery状态的IRF，如果错误的重启了正常工作状态的IRF，会导致合并后的 IRF 仍然处于 Recovery 状态，所有成员设备的业务接口都会被关闭。此时，需要执行 **mad restore** 命令让整个 IRF 系统恢复。

图1-6 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 1-7 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1-7 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 检测和 BFD MAD 检测。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1-1 MAD 检测机制的比较

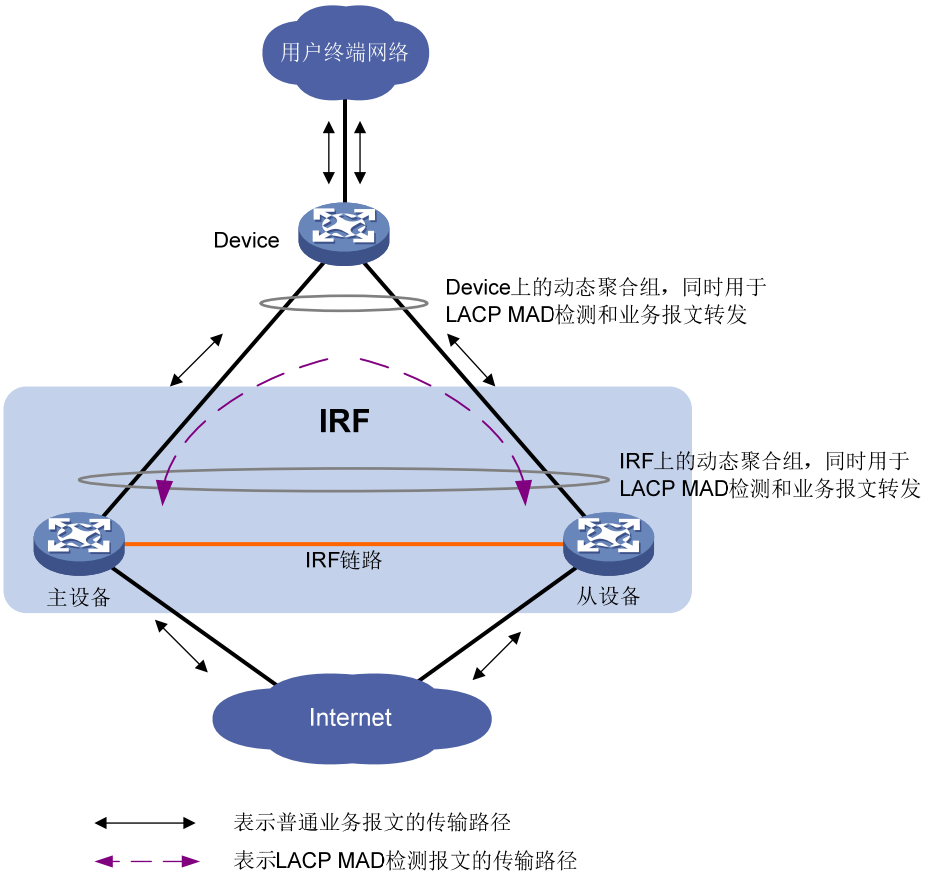
MAD 检测方式	优势	限制	适用组网
LACP MAD	• 检测速度快 • 利用现有聚合组网即可实现，无需占用额外接口	需要使用H3C设备（支持扩展LACP协议报文）作为中间设备	IRF使用聚合链路和上行设备或下行设备连接
BFD MAD	• 检测速度较快 • 使用中间设备时，不要求中间设备必须为 H3C 设备	需要专用的物理链路和三层接口，这些接口不能再传输普通业务流量	• 对组网没有特殊要求 • 如果不使用中间设备，则仅适用于成员设备少（建议仅 2 台成员设备时使用），并且物理距离比较近的组网环境

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 1-8 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1-8 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或管理用以太网口来实现 BFD MAD 检测。

使用管理用以太网口实现BFD MAD时必须使用中间设备(如 图 1-9 所示)，并请注意如下组网要求：

- 每台成员设备都使用管理用以太网口和中间设备建立 BFD MAD 检测链路。

- 为每台成员设备的管理用以太网口配置 MAD IP 地址。

使用以太网端口实现 BFD MAD 时，请注意如下组网要求：

- 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 1-10 所示）。使用中间设备时（如 图 1-9 所示），每台成员设备都需要和中间设备建立BFD MAD检测链路。
- 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1-9 使用中间设备实现 BFD MAD 检测组网示意图

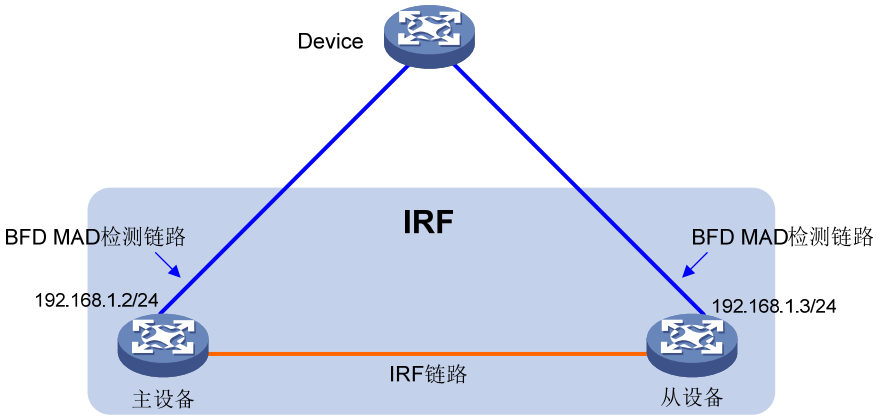
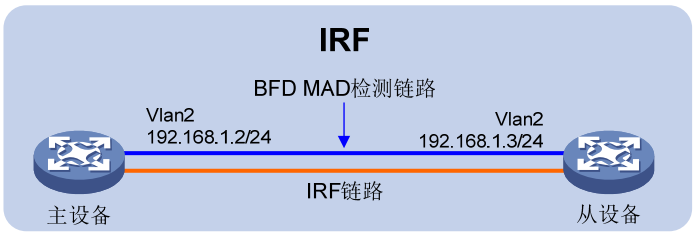


图1-10 不使用中间设备实现 BFD MAD 检测组网示意图



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。

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。

型号	描述
F1000-E-VG/F1000-S-VG	• F1000-E-VG：支持 • F1000-S-VG：不支持
F100-A-G3/F100-C-G3/F100-E-G3/F100-M-G3/F100-S-G3	• F100-A-G3/F100-E-G3：支持 • F100-C-G3/F100-M-G3/F100-S-G3：不支持
F1000-E-G2/F1000-A-G2/F1000-S-G2/F1000-C-G2	支持
F100-E-G2/F100-A-G2/F100-M-G2/F100-S-G2/F100-C-G2	• F100-E-G2/F100-A-G2：支持 • F100-M-G2/F100-S-G2/F100-C-G2：不支持
F1000-C-EI/F100-E-EI/F100-A-EI/F100-C-EI/F100-A-SI	• F1000-C-EI/F100-E-EI/F100-A-EI/F100-A-SI：支持 • F100-C-EI：不支持
F100-A80-WiNet/F100-C80-WiNet/F100-C60-WiNet/F100-C50-WiNet/F100-S80-WiNet	• F100-A80-WiNet：支持 • F100-C80-WiNet/F100-C60-WiNet/F100-C50-WiNet/F100-S80-WiNet：不支持
F1000-C8110/F1000-C8120/F1000-C8130/F1000-C8150/F1000-C8160/F1000-C8170/F1000-C8180	• F1000-C8180/F1000-C8170/F1000-C8160：支持 • F1000-C8150/F1000-C8130/F1000-C8120/F1000-C8110：不支持
F100-C-A3/F100-C-A5/F100-C-A6	不支持
F100-C-A3-W/F100-C-A5-W/F100-C-A6-WL	不支持
F1000-C-HI/F100-C-HI/F100-S-HI/F100-A-HI	• F1000-C-HI/F100-A-HI：支持 • F100-C-HI/F100-S-HI：不支持
F1000-990-AI/F1000-980-AI/F1000-970-AI/F1000-960-AI/F1000-950-AI/F1000-930-AI/F1000-920-AI	支持
LSPM6FWD8/LSQM2FWDSC8	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，必须是相同款型的产品才能组成 IRF。

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。

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。

不能将配置了 Bypass 功能的接口作为 IRF 物理端口。

H3C SecBlade IV NGFW 防火墙插卡仅支持前面板接口作为 IRF 物理端口。

1.3.5 选择连接IRF端口的模块或线缆

不同类型 IRF 物理端口需要采用不同的模块或线缆进行连接：

- 10/100/1000Mbps 千兆以太网口：使用 5 类或 5 类以上双绞线进行连接。
- SFP 口：使用 SFP 光模块及光纤或 SFP 电缆进行连接。
- SFP+口：使用 SFP+光模块及光纤或 SFP+电缆进行连接。

其中双绞线、电缆长度较短，性能和稳定性高，适用于机房内部短距离的 IRF 连接；而光模块和光纤的组合则更加灵活，可以用于较远距离的 IRF 连接。

关于各型号设备上可用于 IRF 连接的模块和电缆，请参见安装手册。

说明

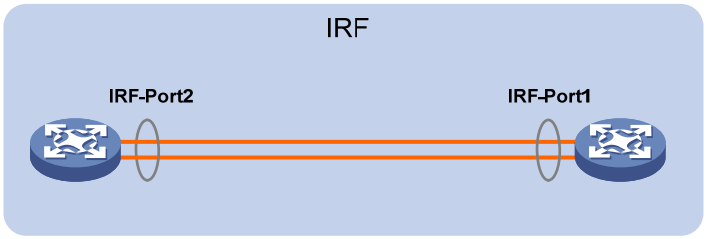
- 有关光模块的详细介绍，请参见《H3C 光模块手册》。
- H3C 光模块的种类随着时间变化有更新的可能性，所以，若您需要准确的模块种类信息，请咨询 H3C 公司市场人员或技术支援人员。

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 1-11 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。可绑定的 IRF 物理端口的最大数目与设备有关，请参见命令手册，且设备仅支持 IRF 物理端口直连组建 IRF，不支持跨中间设备。

图1-11 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“镜像”。

1.3.8 IRF与其他软件特性的兼容性与限制

1. 系统工作模式

在组成 IRF 的所有设备上，系统工作模式的配置（通过 **system-working-mode** 命令配置）必须相同，否则这些设备将无法组成 IRF。关于系统工作模式的介绍，请参见“基础配置指导”中的“设备管理”。

2. 表项容量

在组成 IRF 的所有设备上，表项容量的配置（通过 **hardware-resource switch-mode** 命令配置）必须相同，否则这些设备将无法组成 IRF。关于表项容量的介绍，请参见“基础配置指导”中的“设备管理”。

3. ACL

在组成 IRF 的所有设备上，ACL 硬件模式的相关配置都必须相同，否则这些设备将无法组成 IRF。有关 ACL 硬件模式的详细介绍，请参见“ACL 和 QoS 配置指导”中的“ACL”。

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

(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

  新设备加入 IRF，且新设备的软件版本和主设备的软件版本不一致时，新设备自动从主设备下载启动文件，然后使用新的系统启动文件重启，重新加入 IRF。
- 拆卸IRF物理端口所在的接口模块
- 更换IRF物理端口所在的接口模块

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 配置任务简介

搭建 IRF 配置任务如下：

(1) 分别配置成员编号、成员优先级、IRF 端口。

用户可忽略本步骤，采用快速配置 IRF 基本参数的方式。

a. 配置成员编号

b. （可选）配置成员优先级

c. 配置IRF端口

(2) 快速配置IRF基本参数

用户可忽略本步骤，采用分别配置成员编号、成员优先级、IRF 端口的方式。

(3) 连接IRF物理接口

(4) 访问IRF

1.6.2 配置成员编号

1. 配置限制和指导

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。

IRF 形成后，也可以通过本配置修改成员编号。但是，为了避免配置丢失，形成 IRF 后，尽量不要修改成员编号。

2. 配置步骤

(1) 进入系统视图。

```
system-view
```

(2) 配置成员编号。

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

缺省情况下，设备的成员编号为 1。

1.6.3 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，也可以通过本配置修改成员优先级，但修改不会触发选举，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，设备的成员优先级为 1。

1.6.4 配置IRF端口

1. 配置限制和指导

请先确认哪些接口可以作为IRF物理端口，请参见“1.3.4 确定IRF物理端口”。

同一 IRF 端口绑定的 IRF 物理端口的工作模式必须相同。

将 IRF 物理端口绑定到 IRF 端口后，必须通过 **irf-port-configuration active** 命令手工激活 IRF 端口的配置才能形成 IRF。

系统启动时，通过配置文件将 IRF 物理端口加入 IRF 端口，或者 IRF 形成后再加入新的 IRF 物理端口时，IRF 端口下的配置会自动激活，不需要使用 **irf-port-configuration active** 命令来激活。

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

缺省情况下，设备的接口为开启状态。

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。

(7) 退回系统视图。

quit

(8) 进入 IRF 物理端口视图。

- 进入以太网接口视图。

  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

1.6.5 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

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。

1.6.6 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。设备间将会进行主设备选举，选举失败的一方自动重启。重启完成后，IRF形成。

1.6.7 访问IRF

IRF 的访问方式如下：

- 本地登录：通过任意成员设备的 AUX 或者 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，实际上登录的都是主设备。主设备是 IRF 系统的配置和控制中心，在主设备上配置后，主设备会将相关配置同步给从设备，以便保证主设备和从设备配置的一致性。

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** 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 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，不能通过 **undo shutdown** 命令来激活。

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。

(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VLAN接口进行BFD MAD检测时，请注意 表 1-2 所列配置注意事项。

表1-2 使用 VLAN 接口进行 BFD MAD 检测

注意事项类别	使用限制和注意事项

BFD MAD检测VLAN	• 不允许在 Vlan-interface1 接口上开启 BFD 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 BFD MAD 检测的 VLAN
	  ○ 在IRF 设备和中间设备上，将用于BFD MAD检测的物理接口添加到BFD MAD检测专用 VLAN 中
	  ○ 在 IRF 设备上，创建 BFD MAD 检测的 VLAN 的 VLAN 接口
	• 如果网络中存在多个 IRF，在配置 BFD MAD 时，各 IRF 必须使用不同的 VLAN 作为 BFD MAD 检测专用 VLAN
	• 用于 BFD MAD 检测的 VLAN 接口对应的 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

BFD MAD检测VLAN的特性限制	开启BFD MAD检测功能的VLAN接口及VLAN内的物理端口只能专用于BFD MAD检测，不允许运行其它业务 • 开启 BFD MAD 检测功能的 VLAN 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 • BFD MAD 检测功能与生成树功能互斥，在开启了 BFD MAD 检测功能的 VLAN 接口对应 VLAN 内的端口上，请不要开启生成树协议
BFD 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

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） • 如果网络中存在多个 IRF，在配置 BFD MAD 时，各 IRF 必须使用不同的三层聚合接口做 BFD MAD 检测专用的三层接口 • 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

注意事项类别	使用限制和注意事项
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。中间设备上的端口不用加入聚合组
BFD MAD检测VLAN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
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

使用管理用以太网口进行BFD MAD检测时，请注意 表 1-4 所列配置注意事项。

表1-4 使用管理用以太网口进行 BFD MAD 检测

注意事项类别	使用限制和注意事项
管理用以太网口	将IRF中所有成员设备的管理用以太网口连接到同一台中间设备的普通以太网端口上
BFD MAD检测VLAN	• 将中间设备上与 IRF 成员设备相连的端口配置在一个 VLAN 内（IRF 设备的管理以太网口不需要此配置） • 如果网络中存在多个 IRF，在配置 BFD MAD 时，各 IRF 必须使用不同的 VLAN 作为 BFD MAD 检测专用 VLAN • 请确保中间设备上 BFD MAD 检测 VLAN 中仅包含用于 BFD MAD 检测的端口
MAD IP地址	• 在管理用以太网口使用 **mad ip address** 命令配置 MAD IP 地址，请勿使用 **ip address** 命令配置 • 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。

(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。

3. 使用管理用以太网口进行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。

(3) 进入管理用以太网口的接口视图。

interface m-gigabitethernet *interface-number*

(4) 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。

(5) 为指定成员设备配置 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。

1.7.5 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。

1.8.2 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** | **source-ip** | **source-mac** } *

IRF 链路基于源目的 IP 来进行负载分担。

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** | **source-ip** | **source-mac** } *

缺省情况下，以源目的 IP 进行负载分担。

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。

1.8.3 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“1.8.3　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 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
  ```

本命令缺省情况下 IRF 桥 MAC 保留 6 分钟。

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。

1.8.4 开启启动文件的自动加载功能

1. 功能简介

如果新设备加入 IRF，并且新设备的软件版本和主设备的软件版本不一致，则新加入的设备不能正常启动。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后，再将新设备加入 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，让新设备重新加入 IRF。
- 如果已经开启了启动文件的自动加载功能，则新设备加入 IRF 时，会与主设备的软件版本号进行比较，如果不一致，则自动从主设备下载启动文件，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的文件名与设备上原有启动文件文件名重名，则原有启动文件会被覆盖。

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，请不要手工重启处于加载状态的从设备，否则，会导致该从设备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。

为了能够自动加载成功，请确保从设备存储介质上有足够的空闲空间用于存放新的启动文件。如果从设备存储介质上空闲空间不足，系统会自动删除从设备的当前启动文件来完成加载。如果删除从设备的当前启动文件后空间仍然不足，从设备将无法进行自动加载。此时，需要管理员重启从设备并进入从设备的 Boot ROM 菜单，删除一些不重要的文件后，再让从设备重新加入 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。

1.8.5 拆卸IRF物理端口所在的接口模块

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。

1.8.6 更换IRF物理端口所在的接口模块

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**

操作	命令
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**]

1.10 IRF典型配置举例

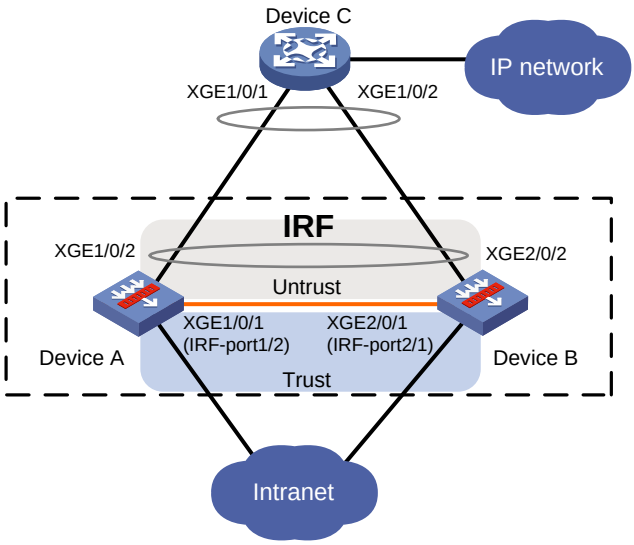
1.10.1 IRF典型配置举例（LACP MAD检测方式）

1. 组网需求

由于公司业务量激增，网络规模迅速扩大，当前中心设备（Device A）安全业务处理能力已经不能满足需求，现在需要另增一台设备Device B，将这两台设备组成一个IRF（如 图 1-12 所示）。由于IRF到中间设备Device C有跨成员设备的聚合链路，且Device C为支持LACP协议的H3C设备，我们配置LACP MAD进行分裂检测。

2. 组网图

图1-12 IRF 典型配置组网图（LACP MAD 检测方式）



3. 配置步骤

(1) 配置 Device A

配置接口 IP 地址、路由、安全域及域间策略保证网络可达，具体配置步骤略。

配置 IRF 中成员编号为 1 的设备的优先级为 32。

```
<DeviceA> system-view
[DeviceA] irf member 1 priority 32
```

配置 IRF 端口 1/2，并将它与物理端口 Ten-GigabitEthernet1/0/1 绑定，并保存配置。

```
[DeviceA] interface ten-gigabitethernet 1/0/1
[DeviceA-Ten-GigabitEthernet1/0/1] shutdown
[DeviceA-Ten-GigabitEthernet1/0/1] quit
```

```
[DeviceA] irf-port 1/2
[DeviceA-irf-port1/2] port group interface ten-gigabitethernet 1/0/1
[DeviceA-irf-port1/2] quit
[DeviceA] interface ten-gigabitethernet 1/0/1
[DeviceA-Ten-GigabitEthernet1/0/1] undo shutdown
[DeviceA-Ten-GigabitEthernet1/0/1] quit
[DeviceA] save
```

激活 IRF 端口下的配置。

```
[DeviceA] irf-port-configuration active
```

(2) 配置 Device B

配置接口 IP 地址、路由、安全域及域间策略保证网络可达，具体配置步骤略。

将 Device B 的成员编号配置为 2，并重启设备使新编号生效。

```
<DeviceB> system-view
[DeviceB] irf member 1 renumber 2
Warning: Renumbering the member ID may result in configuration change or loss. Continue?
[Y/N]:y
[DeviceB] quit
<DeviceB> reboot
```

参照配置组网图进行物理连线。

重新登录到设备，配置 IRF 端口 2/1，将它与物理端口 Ten-GigabitEthernet2/0/1 绑定，并保存配置。

```
<DeviceB> system-view
[DeviceB] interface ten-gigabitethernet 2/0/1
[DeviceB-Ten-GigabitEthernet2/0/1] shutdown
[DeviceB-Ten-GigabitEthernet2/0/1] quit
[DeviceB] irf-port 2/1
[DeviceB-irf-port2/1] port group interface ten-gigabitethernet 2/0/1
[DeviceB-irf-port2/1] quit
[DeviceB] interface ten-gigabitethernet 2/0/1
[DeviceB-Ten-GigabitEthernet2/0/1] undo shutdown
[DeviceB-Ten-GigabitEthernet2/0/1] quit
[DeviceB] save
```

激活 IRF 端口下的配置。

```
[DeviceB] irf-port-configuration active
```

(3) Device A 和 Device B 间将会进行主设备竞选，竞选失败的一方（Device B）将重启，重启完成后，IRF 形成。

(4) 配置 LACP MAD 检测

设置 IRF 域编号为 1。

```
[DeviceA] irf domain 1
```

创建一个动态聚合接口，并开启 LACP MAD 检测功能。

```
[DeviceA] interface route-aggregation 2
[DeviceA-Route-Aggregation2] link-aggregation mode dynamic
[DeviceA-Route-Aggregation2] mad enable
You need to assign a domain ID (range: 0-4294967295)
[Current domain is: 1]:
```

```
The assigned domain ID is: 1
MAD LACP only enable on dynamic aggregation interface.
[DeviceA-Route-Aggregation2] quit
```

在聚合接口中添加成员端口 Ten-GigabitEthernet1/0/2 和 Ten-GigabitEthernet2/0/2，用于 Device A 和 Device B 实现 LACP MAD 检测。

```
[DeviceA] interface ten-gigabitethernet 1/0/2
[DeviceA-Ten-GigabitEthernet1/0/2] port link-aggregation group 2
[DeviceA-Ten-GigabitEthernet1/0/2] quit
[DeviceA] interface ten-gigabitethernet 2/0/2
[DeviceA-Ten-GigabitEthernet2/0/2] port link-aggregation group 2
[DeviceA-Ten-GigabitEthernet2/0/2] quit
```

(5) 配置中间设备 Device C

提示

- Device C 作为中间设备来转发、处理 LACP 协议报文，协助 Device A 和 Device B 进行多 Active 检测。从节约成本的角度考虑，使用一台支持 LACP 协议扩展功能的设备即可。
- 如果中间设备是一个 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。

配置接口 IP 地址、路由保证网络可达，具体配置步骤略。

创建一个动态聚合接口。

```
<DeviceC> system-view
[DeviceC] interface route-aggregation 2
[DeviceC-Route-Aggregation2] link-aggregation mode dynamic
[DeviceC-Route-Aggregation2] quit
```

在聚合接口中添加成员端口 Ten-GigabitEthernet1/0/1 和 Ten-GigabitEthernet1/0/2，用于帮助 LACP MAD 检测。

```
[DeviceC] interface ten-gigabitethernet 1/0/1
[DeviceC-Ten-GigabitEthernet1/0/1] port link-aggregation group 2
[DeviceC-Ten-GigabitEthernet1/0/1] quit
[DeviceC] interface ten-gigabitethernet 1/0/2
[DeviceC-Ten-GigabitEthernet1/0/2] port link-aggregation group 2
[DeviceC-Ten-GigabitEthernet1/0/2] quit
```

4. 验证配置

- IRF 链路正常情况下查看相关配置

查看 IRF 相关信息，可见 IRF 成功建立，且 DeviceA 为主设备。

```
[DeviceA] display irf
MemberID    Role    Priority  CPU-Mac         Description
 *+1        Master  32        487a-da95-93b5  ---
   2        Standby 1         3897-d6a8-1b1a  ---
--------------------------------------------------
 * indicates the device is the master.
 + indicates the device through which the user logs in.

 The bridge MAC of the IRF is: 487a-da95-93b3
```

```
 Auto upgrade                : yes
 Mac persistent              : 6 min
 Domain ID                   : 1
```

查看 LACP MAD 状态，状态正常。

```
[DeviceA] display mad verbose
Multi-active recovery state: No
Excluded ports (user-configured):
Excluded ports (system-configured):
  Ten-GigabitEthernet1/0/1
  Ten-GigabitEthernet2/0/1
MAD ARP disabled.
MAD ND disabled.
MAD LACP enabled interface: Route-Aggregation2
  MAD status              : Normal
  Member ID   Port                                    MAD status
  1           Ten-GigabitEthernet1/0/2                Normal
  2           Ten-GigabitEthernet2/0/2                Normal
MAD BFD disabled.
```

- IRF 链路异常情况下查看相关配置

查看 LACP MAD 状态，状态异常，表示 IRF 分裂。

```
[DeviceA] display mad verbose
Multi-active recovery state: No
Excluded ports (user-configured):
Excluded ports (system-configured):
  Ten-GigabitEthernet1/0/1
MAD ARP disabled.
MAD ND disabled.
MAD LACP enabled interface: Route-Aggregation2
  MAD status              : Faulty
  Member ID   Port                                    MAD status
  1           Ten-GigabitEthernet1/0/2                Faulty
MAD BFD disabled.
```

查看 Device B，可以看到 Device B 上的非保留端口全部被置为 Down，显示信息略。

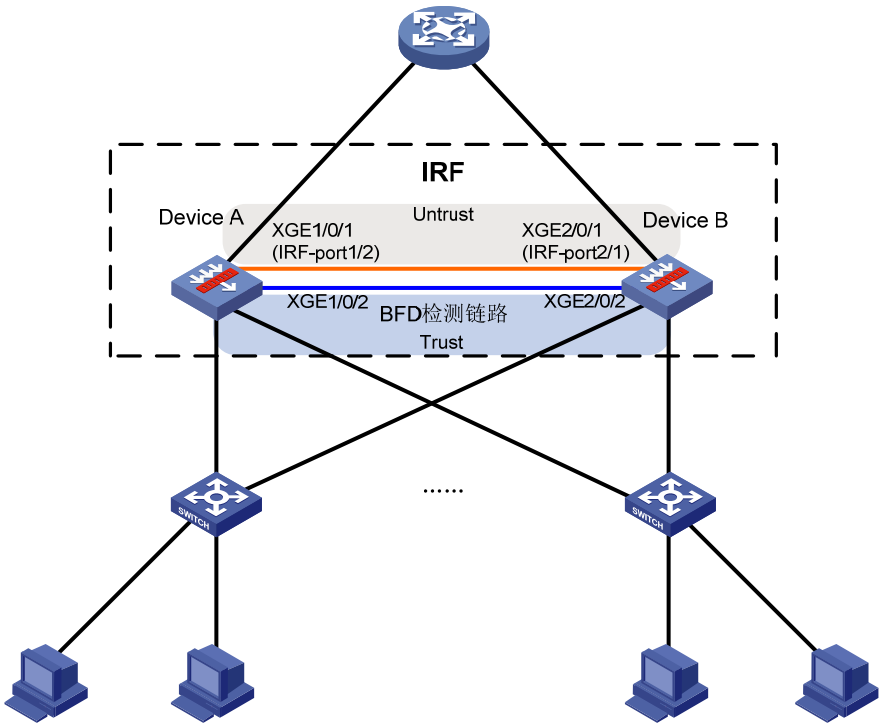
1.10.2 IRF典型配置举例（BFD MAD检测方式）

1. 组网需求

由于网络规模迅速扩大，当前中心设备（Device A）安全业务处理能力已经不能满足需求，现在需要另增一台设备Device B，将这两台设备组成一个IRF（如 图 1-13 所示），并配置BFD MAD进行分裂检测。

2. 组网图

图1-13 IRF 典型配置组网图（BFD MAD 检测方式）



3. 配置步骤

(1) 配置 Device A

配置接口 IP 地址、路由、安全域及域间策略保证网络可达，具体配置步骤略。

配置 IRF 中成员编号为 1 的设备的优先级为 32。

```
<DeviceA> system-view
[DeviceA] irf member 1 priority 32
```

配置 IRF 端口 1/2，并将它与物理端口 Ten-GigabitEthernet1/0/1 绑定，并保存配置。

```
[DeviceA] interface ten-gigabitethernet 1/0/1
[DeviceA-Ten-GigabitEthernet1/0/1] shutdown
[DeviceA-Ten-GigabitEthernet1/0/1] quit
[DeviceA] irf-port 1/2
[DeviceA-irf-port1/2] port group interface ten-gigabitethernet 1/0/1
[DeviceA-irf-port1/2] quit
[DeviceA] interface ten-gigabitethernet 1/0/1
[DeviceA-Ten-GigabitEthernet1/0/1] undo shutdown
[DeviceA-Ten-GigabitEthernet1/0/1] quit
[DeviceA] save
```

激活 IRF 端口下的配置。

```
[DeviceA] irf-port-configuration active
```

(2) 配置 Device B

配置接口 IP 地址、路由、安全域及域间策略保证网络可达，具体配置步骤略。

将 Device B 的成员编号配置为 2，并重启设备使新编号生效。

```
<DeviceB> system-view
[DeviceB] irf member 1 renumber 2
Warning: Renumbering the member ID may result in configuration change or loss. Continue?
[Y/N]:y
[DeviceB] quit
<DeviceB> reboot
```

参照配置组网图进行物理连线。

重新登录到设备，配置 IRF 端口 2/1，并将它与物理端口 Ten-GigabitEthernet2/0/1 绑定，并保存配置。

```
<DeviceB> system-view
[DeviceB] interface ten-gigabitethernet 2/0/1
[DeviceB-Ten-GigabitEthernet2/0/1] shutdown
[DeviceB-Ten-GigabitEthernet2/0/1] quit
[DeviceB] irf-port 2/1
[DeviceB-irf-port2/1] port group interface ten-gigabitethernet 2/0/1
[DeviceB-irf-port2/1] quit
[DeviceB] interface ten-gigabitethernet 2/0/1
[DeviceB-Ten-GigabitEthernet2/0/1] undo shutdown
[DeviceB-Ten-GigabitEthernet2/0/1] quit
[DeviceB] save
```

激活 IRF 端口下的配置。

```
[DeviceB] irf-port-configuration active
```

(3) Device A 和 Device B 间将会进行主设备竞选，竞选失败的一方（Device B）将重启，重启完成后，IRF 形成。

(4) 配置 BFD MAD 检测

创建三层聚合接口 3。

```
[DeviceA] interface route-aggregation 3
[DeviceA-Route-Aggregation3] quit
```

分别将 Device A（成员编号为 1）上的接口 Ten-GigabitEthernet1/0/2 和 Device B（成员编号为 2）上的接口 Ten-GigabitEthernet2/0/2 加入聚合组 3 中。

```
[DeviceA] interface ten-gigabitethernet 1/0/2
[DeviceA-Ten-GigabitEthernet1/0/2] port link-aggregation group 3
[DeviceA-Ten-GigabitEthernet1/0/2] quit
[DeviceA] interface ten-gigabitethernet 2/0/2
[DeviceA-Ten-GigabitEthernet2/0/2] port link-aggregation group 3
[DeviceA-Ten-GigabitEthernet2/0/2] quit
```

开启 BFD MAD 功能，并配置三层聚合接口 3 的 MAD IP 地址。

```
[DeviceA] interface route-aggregation 3
[DeviceA-Route-Aggregation3] mad bfd enable
[DeviceA-Route-Aggregation3] mad ip address 192.168.2.1 24 member 1
[DeviceA-Route-Aggregation3] mad ip address 192.168.2.2 24 member 2
[DeviceA-Route-Aggregation3] quit
```

4. 验证配置

- IRF 链路正常情况下查看相关配置

查看 IRF 相关信息，可见 IRF 成功建立，且 DeviceA 为主设备。

```
[DeviceA] display irf
MemberID    Role    Priority  CPU-Mac         Description
 *+1        Master  32        487a-da95-93b5  ---
   2        Standby 1         3897-d6a8-1b1a  ---
--------------------------------------------------
 * indicates the device is the master.
 + indicates the device through which the user logs in.

 The bridge MAC of the IRF is: 487a-da95-93b3
 Auto upgrade                : yes
 Mac persistent              : 6 min
 Domain ID                   : 0
```

查看 BFD MAD 状态，状态正常。

```
[DeviceA] display mad verbose
Multi-active recovery state: No
Excluded ports (user-configured):
Excluded ports (system-configured):
  Ten-GigabitEthernet1/0/1
  Ten-GigabitEthernet2/0/1
MAD ARP disabled.
MAD ND disabled.
MAD LACP disabled.
MAD BFD enabled interface: Route-Aggregation3
  MAD status                : Normal
  Member ID   MAD IP address       Neighbor   MAD status
  1           192.168.1.1/24       2          Normal
  2           192.168.1.2/24       1          Normal
```

- IRF 链路异常情况下查看相关配置

查看 BFD MAD 状态，状态异常，表示 IRF 分裂。

```
[DeviceA] display mad verbose
Excluded ports (user-configured):
Excluded ports (system-configured):
  Ten-GigabitEthernet1/0/1
MAD ARP disabled.
MAD ND disabled.
MAD LACP disabled.
MAD BFD enabled interface: Route-Aggregation3
  MAD status                : Faulty
  Member ID   MAD IP address       Neighbor   MAD status
  1           192.168.1.1/24       2          Faulty
```

查看 Device B，可以看到 Device B 上的非保留端口全部被置为 Down，显示信息略。

目 录

1 Context

1.1 Context简介

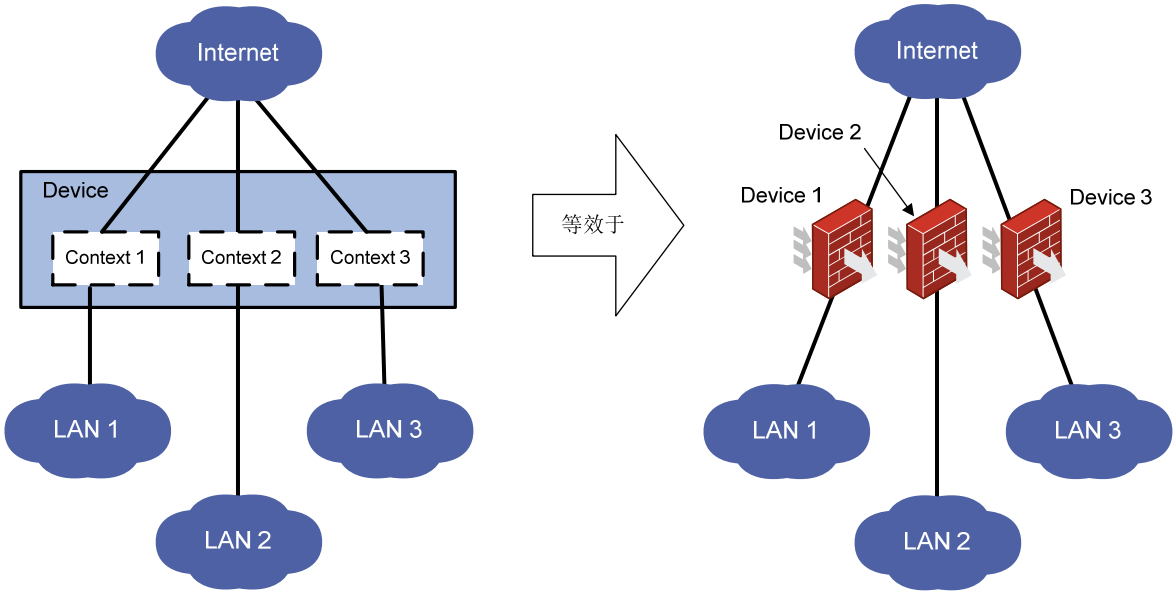
通过虚拟化技术将一台物理设备划分成多台逻辑设备，每台逻辑设备就称为一个 Context。每个 Context 拥有自己专属的软硬件资源，独立运行。

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，没有分配的 CPU 资源/磁盘/内存空间、接口、VLAN 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。
- 非缺省 Context 下不支持共享口的报文捕获功能，关于报文捕获功能的详细描述请参见“网络管理和监控配置指导”中的“报文捕获配置”。

1.2 Context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。

型号	说明
F1000-E-VG/F1000-S-VG	• F1000-E-VG：支持 • F1000-S-VG：不支持
F100-A-G3/F100-C-G3/F100-E-G3/F100-M-G3/F100-S-G3	• F100-A-G3/F100-E-G3：支持 • F100-C-G3/F100-M-G3/F100-S-G3：不支持
F1000-E-G2/F1000-A-G2/F1000-S-G2/F1000-C-G2	支持
F100-E-G2/F100-A-G2/F100-M-G2/F100-S-G2/F100-C-G2	• F100-E-G2/F100-A-G2：支持 • F100-M-G2/F100-S-G2/F100-C-G2：不支持
F1000-C-EI/F100-E-EI/F100-A-EI/F100-C-EI/F100-A-SI	• F1000-C-EI/F100-E-EI/F100-A-EI/F100-A-SI：支持 • F100-C-EI：不支持
F100-A80-WiNet/F100-C80-WiNet/F100-C60-WiNet/F100-C50-WiNet/F100-S80-WiNet	• F100-A80-WiNet：支持 • F100-C80-WiNet/F100-C60-WiNet/F100-C50-WiNet/F100-S80-WiNet：不支持
F1000-C8110/F1000-C8120/F1000-C8130/F1000-C8150/F1000-C8160/F1000-C8170/F1000-C8180	• F1000-C8160/F1000-C8170/F1000-C8180：支持 • F1000-C8110/F1000-C8120/F1000-C8130/F1000-C8150：不支持
F100-C-A3/F100-C-A5/F100-C-A6	不支持
F100-C-A3-W/F100-C-A5-W/F100-C-A6-WL	不支持
F1000-C-HI/F100-C-HI/F100-S-HI/F100-A-HI	• F1000-C-HI/F100-A-HI：支持 • F100-C-HI/F100-S-HI：不支持
F1000-990-AI/F1000-980-AI/F1000-970-AI/F1000-960-AI/F1000-950-AI/F1000-930-AI/F1000-920-AI	支持
LSPM6FWD8/LSQM2FWDSC8	支持

1.3 Context配置限制和指导

对于本文列出的命令，缺省Context均支持，非缺省Context只支持**display context interface**命令。

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。

1.4 Context配置任务简介

Context 配置任务如下：

(1) 创建Context

(2) （可选）为Context分配接口和VLAN资源

- 为Context分配接口
- 为Context分配VLAN

(3) （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

(4) 启动Context

(5) （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

(6) 访问和管理Context

(7) （可选）收集各Context的日志信息

1.5 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。

- 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。

1.6 为Context分配接口和VLAN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 **shutdown**、**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。

1.7 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。

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 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。

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。

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。

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。

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。

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. 配置限制和指导

在使用 **context start force** 前，用户可以通过 **display context**、**display system internal context configuration-status**、**display system internal context id contex-id running-status** 命令查看 Context 的运行情况。

3. 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。

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

缺省情况下，Context 可以使用物理设备上的所有空闲磁盘空间。

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。

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

缺省情况下，所有 Context 共享物理设备上的所有内存空间，每个 Context 可使用的内存空间上限为空闲内存空间值。

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的阈值后，发往此 Context 的广播报文会被设备丢弃，否则不会被丢弃。

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。当广组播报文总速率和单个 Context 广组播报文速率均达到各自的阈值后，发往此 Context 的广组播报文会被设备丢弃，否则不会被丢弃。

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关闭状态。

1.12 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*

1.13 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。

表1-1 Context 显示和维护

操作	命令
显示Context的相关信息	**display context** [**name** *context-name*] [**verbose**]
显示Context内可分配业务资源的使用情况	**display context** [**name** *context-name*] **capability** [**security-policy** | **session** [**slot** *slot-number*] | **sslvpn-user**]
显示Context入方向广播报文的速率限制信息	**display context name** *context-name* **capability inbound broadcast slot** *slot-number*
显示Context入方向组播报文的速率限制信息	**display context name** *context-name* **capability inbound multicast slot** *slot-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	**display context** [**name** *context-name*] **resource** [**cpu** | **disk** | **memory**] [**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**
清除Context入方向广播报文的速率限制信息	**reset context name** *context-name* **capability inbound broadcast slot** *slot-number*
清除Context入方向组播报文的速率限制信息	**reset context name** *context-name* **capability inbound multicast slot** *slot-number*

1.14 Context典型配置举例

1.14.1 Context基本组网配置举例

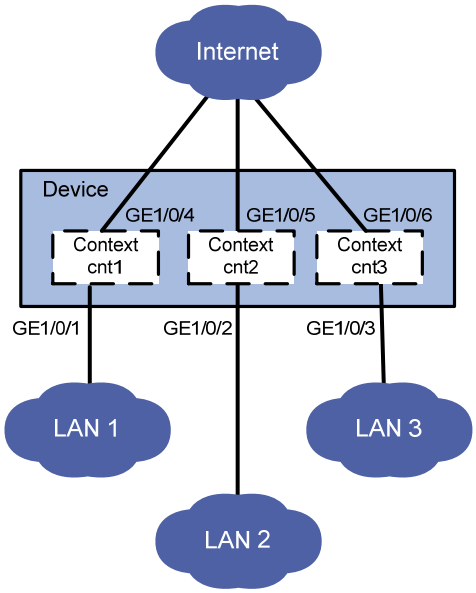
1. 组网需求

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1-2 Context 基本组网配置组网图



3. 配置步骤

(1) 创建并配置 Context cnt1，供公司 A 使用

创建 Context cnt1，设置描述信息。

```
<Device> system-view
[Device] context cnt1
[Device-context-2-cnt1] description context-1
```

配置 Context cnt1 的磁盘使用上限为 60%。

```
[Device-context-2-cnt1] limit-resource disk slot 1 cpu 0 ratio 60
```

配置 Context cnt1 的内存使用上限为 60%。

```
[Device-context-2-cnt1] limit-resource memory slot 1 cpu 0 ratio 60
```

配置 Context cnt1 的 CPU 权重为 8。

```
[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
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) 2004-2018 New H3C Technologies Co., Ltd. All rights reserved.*
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************

<Device> 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。

```
[Device] 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]
```

(2) 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
```

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
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) 2004-2018 New H3C Technologies Co., Ltd. All rights reserved.*
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************

<Device> 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。

```
[Device] 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]
```

(3) 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
```

#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
Configuration of the interfaces will be lost. Continue? [Y/N]:y
```

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) 2004-2018 New H3C Technologies Co., Ltd. All rights reserved.*
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************
```

```
<Device> 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。

```
[Device] 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) 2004-2018 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
```

显示信息略……。