



配置指南-网管与监控

本分册介绍网管与监控配置指南相关内容，包括以下章节：

1. SNMP
2. NETCONF
3. RMON
4. NTP
5. SNTP
6. SPAN-RSPAN
7. ERSPAN
8. sFlow

1 SNMP

1.1 概述

SNMP 是 Simple Network Management Protocol (简单网络管理协议) 的缩写, 在 1988 年 8 月就成为一个网络管理标准 RFC1157。到目前, 因众多厂家对该协议的支持, SNMP 已成为事实上的网管标准, 适合于在多厂家系统的互连环境中使用。利用 SNMP 协议, 网络管理员可以对网络上的节点进行信息查询、网络配置、故障定位、容量规划, 网络监控和管理是 SNMP 的基本功能。

📄 SNMP 协议版本

目前 SNMP 支持以下版本:

- SNMPv1 : 简单网络管理协议的第一个正式版本, 在 RFC1157 中定义。
- SNMPv2C : 基于共同体 (Community-Based) 的 SNMPv2 管理架构, 在 RFC1901 中定义。
- SNMPv3 : 通过对数据进行鉴别和加密, 提供了以下的安全特性:
 1. 确保数据在传输过程中不被篡改;
 2. 确保数据从合法的数据源发出;
 3. 加密报文, 确保数据的机密性。

协议规范

- RFC 1157 , Simple Network Management Protocol (SNMP)
- RFC 1901 , Introduction to Community-based SNMPv2
- RFC 2578 , Structure of Management Information Version 2 (SMIv2)
- RFC 2579 , Textual Conventions for SMIv2
- RFC 3411 , An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks
- RFC 3412 , Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)
- RFC 3413 , Simple Network Management Protocol (SNMP) Applications
- RFC 3414 , User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)
- RFC 3415 , View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)
- RFC 3416 , Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)
- RFC 3417 , Transport Mappings for the Simple Network Management Protocol (SNMP)
- RFC 3418 , Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)
- RFC 3419 , Textual Conventions for Transport Addresses

1.2 典型应用

典型应用	场景描述
通过 SNMP 管理网络设备	通过 SNMP 网络管理器对网络设备进行管理和监控。

1.2.1 通过 SNMP 管理网络设备

应用场景

以下图为例，用户通过 SNMP 网络管理器，来对网络设备 A 进行管理和监控。

图 1-1



【注释】 A 为需要被管理的网络设备。
PC 为网络管理站。

功能部署

网络管理站和被管理的网络设备通过网络连接，用户在网络管理站上，通过 SNMP 网络管理器，访问网络设备上的管理信息数据库，以及接收来自网络设备主动发出的消息，来对网络设备进行管理和监控。

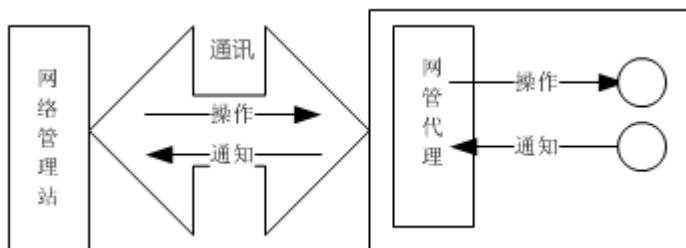
1.3 功能详解

基本概念

SNMP 是一个应用层协议，为客户机/服务器模式，包括三个部分：

- SNMP 网络管理器
- SNMP 代理
- MIB 管理信息库

图 1-2 网络管理站（NMS）与网管代理（Agent）的关系图



SNMP 网络管理器

SNMP 网络管理器，是采用 SNMP 来对网络进行控制和监控系统，也称为 NMS (Network Management System)。

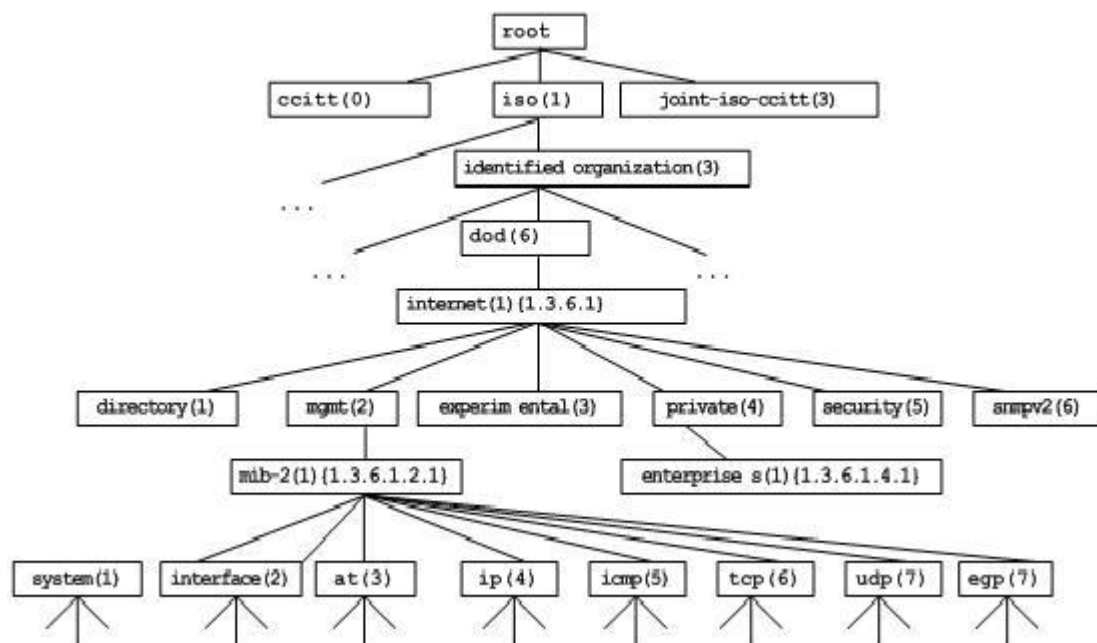
SNMP 代理

SNMP 代理 (SNMP Agent, 下文简称为 Agent) 是运行在被管理设备上的软件，负责接受、处理并且响应来自 NMS 的监控和控制报文，也可以主动发送一些消息报文给 NMS。

MIB

MIB (Management Information Base) 是一个虚拟的网络管理信息库。被管理的网络设备中包含大量信息，为了能在 SNMP 报文中唯一的标识某个特定的管理单元，MIB 采用树形层次结构来描述，树的节点表示某个特定的管理单元。为了唯一标识网络设备中的某个管理单元 System，可以采用一串的数字来表示，MIB 则是网络设备的单元标识符的集合。

图 1-3 MIB 树形层次结构



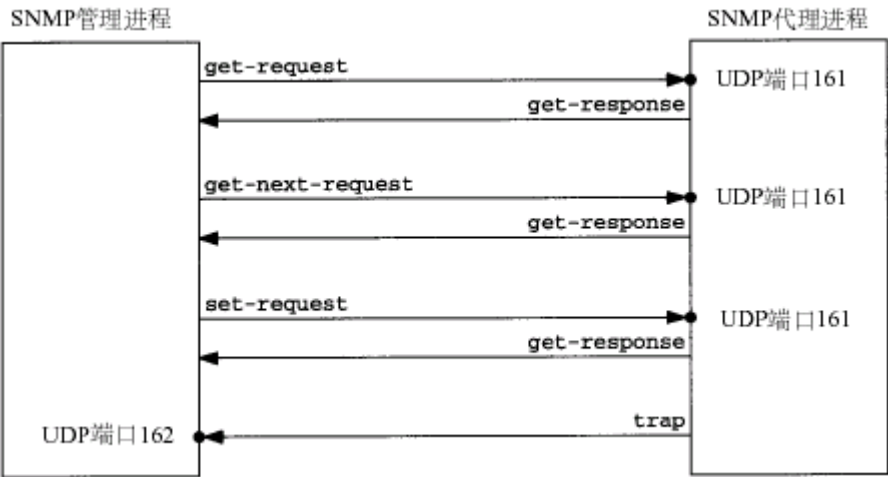
操作类型

SNMP 协议中的 NMS 和 Agent 之间的交互信息，定义了 6 种操作类型：

- Get-request 操作：NMS 从 Agent 提取一个或多个参数值。
- Get-next-request 操作：NMS 从 Agent 提取一个或多个参数的下一个参数值。
- Get-bulk 操作：NMS 从 Agent 提取批量的参数值；
- Set-request 操作：NMS 设置 Agent 的一个或多个参数值。
- Get-response 操作：Agent 返回的一个或多个参数值，是 Agent 对 NMS 前面 3 个操作的响应操作。
- Trap 操作：Agent 主动发出的报文，通知 NMS 有某些事情发生。

前面的 4 个报文是由 NMS 向 Agent 发出的，后面两个是 Agent 发给 NMS 的（注意：SNMPv1 版本不支持 Get-bulk 操作）。
下图描述了这几种操作。

图 1-4 SNMP 的报文类型



NMS 向 Agent 发出的前面 3 种操作和 Agent 的应答操作采用 UDP 的 161 端口。Agent 发出的 Trap 操作采用 UDP 的 162 端口。

功能特性

功能特性	作用
SNMP 基本功能	配置网络设备上的 SNMP 代理，实现对网络上的节点进行信息查询、网络配置、故障定位、容量规划等基本功能。
SNMPv1 及 SNMPv2C	采用基于共同体的安全架构，包括认证名和访问权限。
SNMPv3	SNMPv3 重新定义了 SNMP 架构，主要是在安全功能上进行了增强，包括支持基于用户的安全模型，以及支持基于视图的访问控制模型等。SNMPv3 架构内已经包含了 SNMPv1 和 SNMPv2C 所有的功能。

1.3.1 SNMP 基本功能

工作原理

📌 工作过程

SNMP 协议交互是应答式的（报文交互参见图 1-4 SNMP 的报文类型）。NSM 向 Agent 主动发起请求，包括 Get-request、Get-next-request、Get-bulk 和 Set-request，Agent 接收请求并完成操作后以 Get-response 作为应答。Agent 有时候也会向 NSM 主动发出 Trap 和 Inform 消息，其中 Trap 消息不需要应答，而 Inform 消息则需要 NSM 回送一个 Inform-response 应答，表示收到消息，否则 Agent 将会重发 Inform 消息。

相关配置

📌 屏蔽或关闭 SNMP 代理

缺省时启动 SNMP 功能。

使用 **no snmp-server** 命令屏蔽 SNMP 代理功能。

执行 **no enable service snmp-agent** 命令，直接关闭 SNMP 所有服务。

📌 设置 SNMP 基本参数

缺省时系统联系方式、系统位置和设备的网元信息为空；序列号缺省值是 60FF60；缺省最大数据报文长度 1572 字节；缺省的 SNMP 服务 UDP 端口号是 161。

使用 **snmp-server contact** 命令配置或删除系统联系方式。

使用 **snmp-server location** 命令配置或删除系统位置。

使用 **snmp-server chassis-id** 命令配置系统序列码或恢复缺省值。

使用 **snmp-server packet-size** 命令配置代理最大数据报文长度或恢复缺省值。

使用 **snmp-server net-id** 命令配置或删除设备的网元信息。

使用 **snmp-server udp-port** 命令设置 SNMP 服务 UDP 端口号或恢复缺省值。

缺省时 v3 版本功能，关闭 SNMP v1 和 SNMP v2c 版本功能。

使用 **no snmp-server enable version [v1|v2c|v3]** 命令可以单独关闭指定的 SNMP 协议版本功能。

📌 配置 SNMP 主机地址

缺省情况下，没有 SNMP 主机。

使用 **snmp-server host** 命令配置 Agent 主动发送消息的 NMS 主机地址或删除指定 SNMP 主机地址。发给主机的消息可以绑定 SNMP 的版本、接收端口、认证名或用户。该命令与 **snmp-server enable traps** 命令一起使用，主动给 NMS 发送 Trap 消息。

📌 设置 Trap 消息参数

缺省情况下，禁止 SNMP 向 NMS 主动发送 Trap 消息；打开接口发送 Link Trap 功能；关闭发送系统重启 Trap 功能；Trap 消息缺省不带私有字段。

缺省时，SNMP 报文从哪个接口出去，就使用哪个接口的 IP 地址作为源地址。

缺省时 Trap 消息报文的队列长度为 10，发送 Trap 消息的时间间隔为 30 秒。

使用 **snmp-server enable traps** 命令配置 Agent 主动或禁止向 NMS 发送 Trap 消息。

使用 **snmp trap link-status** 命令打开或关闭接口发送 Link Trap 功能。

使用 **snmp-server trap-source** 命令指定发送消息的源地址或恢复缺省值。

使用 **snmp-server queue-length** 命令设置 Trap 消息报文的队列长度或恢复缺省值。

使用 **snmp-server trap-timeout** 命令设置发送 Trap 消息的时间间隔或恢复缺省值。

使用 **snmp-server trap-format private** 命令设置或关闭发送 Trap 消息时携带私有字段的功能。

使用 **snmp-server system-shutdown** 命令打开或关闭发送系统重启 Trap 功能。

📌 设置 SNMP 攻击防护检测功能

缺省情况下，没有打开 SNMP 攻击防护检测功能。

使用 **snmp-server authentication attempt times exceed { lock | lock-time minutes | unlock }** 命令设置打开攻击防护检测功能。

1.3.2 SNMPv1 及 SNMPv2C

SNMPv1 和 SNMPv2C 都采用基于共同体(Community-based)的安全架构。通过定义主机地址以及认证名(Community String)来限定能够对代理的 MIB 进行操作的管理者。

工作原理

SNMPv1 和 SNMPv2 版本使用认证名来鉴别是否有权使用 MIB 对象。为了能够管理设备，网络管理系统 (NMS)的认证名必须同设备中定义的某个认证名一致。

SNMPv2C 增加了 Get-bulk 操作机制并且能够对管理工作站返回更加详细的错误信息类型。Get-bulk 操作能够一次性地获取表格中的所有信息或者获取大批量的数据，从而减少请求-响应的次数。SNMPv2C 错误处理能力的提高包括扩充错误代码以区分不同类型的错误，而在 SNMPv1 中这些错误仅有一种错误代码。现在通过错误代码可以区分错误类型。由于网络上可能同时存在支持 SNMPv1 和 SNMPv2C 的管理工作站，因此 SNMP 代理必须能够识别 SNMPv1 和 SNMPv2C 报文，并且能返回相应版本的报文。

📌 安全

一个认证名有以下属性：

- 只读(Read-only)：为被授权的管理工作站提供对所有 MIB 变量的读权限。

- 读写(Read-write)：为被授权的管理工作站提供对所有 MIB 变量的读写权限。

相关配置

设置认证名及访问权限

所有认证名的缺省访问权限为只读。

使用 **snmp-server community** 命令配置或删除认证名和访问权限。

该命令为启用设备 SNMP 代理功能的第一个重要命令，指定了团体的属性、允许访问 MIB 的 NMS 范围等等。

1.3.3 SNMPv3

SNMPv3 重新定义了 SNMP 架构，将之前的 SNMPv1 和 SNMPv2 的功能也纳入到 SNMPv3 体系中。

工作原理

网络管理系统 (NMS) 和 SNMP 代理 (SNMP Agent) 都称为 SNMP 实体。在 SNMPv3 架构中，SNMP 实体分为引擎和应用两大部分，其中 SNMP 引擎用于发送和接收信息、鉴定和加密信息以及对管理对象的控制访问。SNMP 应用指的是 SNMP 内部的应用程序，利用 SNMP 引擎提供的服务进行工作。

SNMPv3 版本使用基于用户的安全模型 (USM) 来鉴别是否有权使用 MIB 对象。为了能够管理设备，网络管理系统 (NMS) 的用户和安全级别必须同设备中定义的某个 SNMP 用户一致。

SNMPv3 版本规定 NSM 在管理设备的时候，必须先得知设备上 SNMP Agent 的引擎标识。SNMPv3 定义了 Discover 和 Report 操作机制，NSM 在不知道 Agent 引擎标识的情况下，可以先向 Agent 发送 Discover 报文，而 Agent 以 Report 响应，并在响应报文中携带了引擎标识信息。此后，NSM 和 Agent 之间的管理操作必须携带该引擎标识。

安全

- SNMPv3 通过安全模型以及安全级别来确定对数据采用哪种安全机制进行处理。目前可用的安全模型有三种类别：SNMPv1、SNMPv2C、SNMPv3。SNMPv3 将 SNMPv1 和 SNMPv2C 也纳入到安全模型中。

SNMPv1 及 SNMPv2C 安全模型和级别

安全模型	安全级别	鉴别	加密	说明
SNMPv1	noAuthNoPriv	认证名	无	通过认证名确认数据的合法性
SNMPv2c	noAuthNoPriv	认证名	无	通过认证名确认数据的合法性

SNMPv3 安全模型以及安全级别

安全模型	安全级别	鉴别	加密	说明
SNMPv3	noAuthNoPriv	用户名	无	通过用户名确认数据的合法性
SNMPv3	authNoPriv	MD5 或者 SHA	无	提供基于 HMAC-MD5 或者 HMAC-SHA 的数据鉴别机制
SNMPv3	authPriv	MD5 或者 SHA	DES	提供基于 HMAC-MD5 或者 HMAC-SHA 的数据鉴别机制提供基于 CBC-DES 的数据加密机制

引擎标识

引擎标识用于唯一标识一个 SNMP 引擎。由于每个 SNMP 实体仅包含一个 SNMP 引擎，它将在一个管理域中唯一标识一个 SNMP 实体。因此，作为一个实体的 SNMPv3 代理必须拥有一个唯一的引擎标识，即 SnmpEngineID。

引擎标识为一个 OCTET STRING，长度为 5~32 字节长。在 RFC3411 中定义了引擎标识的格式：

- 前 4 个字节标识厂商的私有企业号（由 IANA 分配），用 HEX 表示。
- 第 5 个字节表示剩下的字节如何标识：
- 0：保留
- 1：后面 4 个字节是一个 Ipv4 地址。
- 2：后面 16 个字节是一个 Ipv6 地址。
- 3：后面 6 个字节是一个 MAC 地址。
- 4：文本，最长 27 个字节，由厂商自行定义。
- 5：16 进制值，最长 27 个字节，由厂商自行定义。
- 6-127：保留。
- 128-255：由厂商特定的格式。

相关配置

配置 MIB 视图和组

缺省配置一个 default 视图，允许访问所有的 MIB 对象。

缺省没有配置用户组。

使用 **snmp-server view** 命令配置或删除视图；使用 **snmp-server group** 命令配置或删除用户组。

可以配置一条或者多条指令，来指定多个不同的共同体名称，使得网络设备可以供不同的权限的 NMS 的管理。

配置 SNMP 用户

缺省没有配置用户。

配置 **snmp-server user** 命令配置或删除用户。

NMS 只有使用合法的用户才能同代理进行通信。

对于 SNMPv3 用户，可以指定安全级别（是否需要进行认证、是否需要进行加密等）、认证算法（MD5 或 SHA）、认证口令、加密算法（目前只有 DES）和加密口令。

1.4 配置详解

配置项	配置建议 & 相关命令	
配置 SNMP 基本功能	 必须配置。使用户可以通过 NMS 访问 Agent。	
	enable service snmp-agent	启动 Agent 功能。
	snmp-server community	配置认证名和访问权限。
	snmp-server user	配置 SNMP 用户信息。
	snmp-server view	配置 SNMP 视图。
	snmp-server group	配置 SNMP 用户组。
	snmp-server authentication	配置 SNMP 攻击防护检测功能
启用 Trap 功能	 可选配置。使 Agent 主动向 NMS 发送 Trap 消息。	
	snmp-server host	配置 NMS 主机地址。
	snmp-server enable traps	Agent 主动向 NMS 发送 Trap 消息。
	snmp trap link-status	打开接口发送 Link Trap 功能。
	snmp-server system-shutdown	打开发送系统重启 Trap 功能。
	snmp-server trap-source	指定发送 Trap 消息的源地址。
	snmp-server trap-format private	发送 Trap 消息时携带私有字段
屏蔽 Agent 功能	 可选配置。在不需要 Agent 服务的时候，屏蔽 Agent 功能。	
	no snmp-server	屏蔽 Agent 功能。
设置 SNMP 控制参数	 可选配置。用于设置或修改 SNMP 控制参数。	
	snmp-server contact	设置设备的联系方式。
	snmp-server location	设置设备位置。
	snmp-server chassis-id	设置设备序列码。
	snmp-server net-id	设置设备的网元信息。
	snmp-server packet-size	修改最大数据报文长度。
	snmp-server udp-port	修改 SNMP 服务 UDP 端口号。
	snmp-server queue-length	修改 Trap 消息报文的队列长度。
	snmp-server trap-timeout	修改发送 Trap 消息的时间间隔。
	no snmp-server enable version	关闭指定的 SNMP 协议版本

1.4.1 配置 SNMP 基本功能

配置效果

使用户可以通过 NMS 访问 Agent。

注意事项

- 网络设备上默认没有设置认证名，无法使用 SNMPv1 或 SNMPv2C 访问网络设备的 MIB。设置认证名时，如果没有指定访问权限，则默认的访问权限是只读（Read-only）。

配置方法

▾ 配置 SNMP 视图

- 可选配置。
- 使用基于视图的访问控制（VACM）功能时需要进行配置。

▾ 配置 SNMP 用户组

- 可选配置。
- 使用基于视图的访问控制（VACM）功能时需要进行配置。

▾ 配置认证名和访问权限

- 必选配置。
- 使用 SNMPv1 和 SNMPv2C 管理网络设备必须在 agent 设备上设置认证名。

▾ 配置 SNMP 用户信息

- 必选配置。
- 使用 SNMPv3 管理网络设备必须设置用户。

▾ 启动 Agent 功能

- 可选配置。
- 默认开启 Agent 功能，在 Agent 功能关闭后需要再次开启时，须使用此命令。

▾ 打开 SNMP 攻击防护检测功能

- 可选配置。
- 默认关闭 SNMP 攻击防护检测功能，在需要防止恶意攻击时，在 agent 上使用该项配置。

检验方法

使用 **show snmp** 命令查看设备上的 snmp 功能。

相关命令

▾ 配置 SNMP 视图

【命令格式】 **snmp-server view** *view-name oid-tree* { **include** | **exclude** }

【参数说明】 *view-name* : 视图名。

oid-tree : 视图关联的 MIB 对象, 是一棵 MIB 子树。

include : 标明该 MIB 对象子树被包含在视图之内。

exclude : 标明该 MIB 对象子树被排除在视图之外。

【命令模式】 全局配置模式

【使用指导】 指定视图的名称, 用于基于视图的管理。

配置 SNMP 用户组

【命令格式】 **snmp-server group** *groupname* { **v1** | **v2c** | **v3** { **auth** | **noauth** | **priv** } } [**read** *readview*] [**write** *writeview*] [**access** { **ipv6** *ipv6-aclname* | *aclnum* | *aclname* }]

【参数说明】 **v1** | **v2c** | **v3** : 指明 SNMP 版本。

auth : 该组的用户传输的消息需要验证但数据不需要保密, 只对 v3 有效。

noauth : 该组用户传输的消息不需要验证数据也不需要保密, 只对 v3 有效。

priv : 该组用户传输的消息需要验证同时传输的数据需要保密, 只对 v3 有效。

readview : 关联一个只读的视图。

writeview : 关联一个读写视图。

aclnum : 访问列表序列号, 关联指定的访问列表, 指定能访问 MIB 的 ipv4 NMS 地址范围。

aclname : 访问列表名称, 关联指定的访问列表, 指定能访问 MIB 的 ipv4 NMS 地址范围。

ipv6-aclname : ipv6 访问列表名称, 关联指定的访问列表, 指定能访问 MIB 的 ipv6 NMS 地址范围。

【命令模式】 全局配置模式

【使用指导】 将某些用户和一个组关联, 再将某个组与某个视图关联。一个组内的用户具有相同的访问权限。通过这种方式判定操作关联的管理对象是否在视图允许之内, 只有在视图允许之内的管理对象才被允许访问。

配置认证名和访问权限

【命令格式】 **snmp-server community** [*0* | *7*] *string* [**view** *view-name*] [[**ro** | **rw**] [**host** *ipaddr*]] [**ipv6** *ipv6-aclname*] [*aclnum* | *aclname*]

【参数说明】 *0* : 表示输入的团体字符串为明文字符串。

7 : 表示输入的团体字符串为密文字符串。

string : 团体字符串, 相当于 NMS 和 SNMP 代理之间的通信密码。

view-name : 指定视图的名称, 用于基于视图的管理。

ro : 指定 NMS 对 MIB 的变量只能读, 不能修改。

rw : NMS 对 MIB 的变量可读可写。

aclnum : 访问列表序列号, 关联指定的访问列表, 指定能访问 MIB 的 ipv4 NMS 地址范围。

aclname : 访问列表名称, 关联指定的访问列表, 指定能访问 MIB 的 ipv4 NMS 地址范围。

ipv6-aclname : ipv6 访问列表名称, 关联指定的访问列表, 指定能访问 MIB 的 ipv6 NMS 地址范围。

ipaddr : 关联 NMS 地址, 指定访问 MIB 的 NMS 地址。

【命令模式】 全局配置模式

【使用指导】 该命令为启用设备 SNMP 代理功能的第一个重要命令, 指定了团体的属性、允许访问 MIB 的 NMS 范围等等。要关闭 SNMP 代理功能, 执行 **no snmp-server** 命令即可。

配置 SNMP 用户

【命令格式】 **snmp-server user** *username groupname* { **v1** | **v2c** | **v3** [**encrypted**] [**auth** { **md5** | **sha** } *auth-password*] [**priv** **des56** *priv-password*] } [**access** { **ipv6** *ipv6-aclname* | *aclnum* | *aclname* }]

【参数说明】 *username* : 用户名。

groupname : 该用户对应的组名。

v1 | **v2c** | **v3** : 指明 SNMP 版本。只有 v3 支持后面的安全参数。

encrypted : 指定的是密码输入的方式为密文输入。否则, 以明文输入。如果选择了以密文输入, 则需要输入连续的 16 进制数字字符表示的密钥。注意使用 MD5 的认证密钥长度为 16 字节, 而 SHA 认证协议密钥长度为 20 字节。以两个字符表示一个字节。加密表示的密钥仅对本引擎有效。

auth : 指定是否使用验证。

md5 : 指定使用 MD5 认证协议。**sha** 指定使用 SHA 认证协议。

auth-password : 配置认证协议使用的口令字符串 (不超过 32 个字符)。系统将这些口令转换成相应的认证密钥。

priv : 指定是否使用保密。**des56** 指明使用 56 位的 DES 加密协议。

priv-password : 为加密用的口令字符串 (不超过 32 个字符)。系统将这个口令转换成相应的加密密钥。

aclnum : 访问列表序列号, 关联指定的访问列表, 指定能访问 MIB 的 ipv4 NMS 地址范围。

aclname : 访问列表名称, 关联指定的访问列表, 指定能访问 MIB 的 ipv4 NMS 地址范围。

ipv6-aclname : ipv6 访问列表名称, 关联指定的访问列表, 指定能访问 MIB 的 ipv6 NMS 地址范围。

【命令模式】 全局配置模式

【使用指导】 配置用户的信息, 以使 NMS 使用合法的用户同代理进行通信。

对于 SNMPv3 用户, 可以指定安全级别、认证算法 (MD5 或 SHA)、认证口令、加密算法 (目前只有 DES) 和加密口令。

启动 Agent 功能

【命令格式】 **enable service snmp-agent**

【参数说明】

【配置模式】 特权用户模式

【使用指导】 该命令用于启动设备的 SNMP 代理功能。

启动 SNMP 攻击防护检测功能

【命令格式】 **snmp-server authentication attempt times exceed** { **lock** | **lock-time minutes** | **unlock** }

【参数说明】 *times* : 连续认证失败的尝试次数。

lock : 连续认证失败后, 永久禁止该源 IP 地址重新进行认证访问, 需要管理员手工解除。

lock-time minutes : 连续认证失败后, 禁止该源 IP 地址一段时间, 超过限定时间后可以重新进行认证访问。

unlock : 连续认证失败后, 允许该源 IP 地址继续进行访问, 相关于没有配置 SNMP 的攻击防护检测功能。

【命令模式】 全局配置模式

【使用指导】 配置 SNMP 攻击防护检测功能, 以使在连续认证失败后做出对应的处理策略。

对于被永久禁止的源 IP 地址, 只有管理员进行手工解除后, 该源 IP 地址才能重新访问认证。

对于被禁止一段时间内认证的源 IP 地址, 当设置的禁止时间超时或者管理员手工解除后该源 IP 地址才能重新访问认证。

显示 SNMP 的状态信息

【命令格式】 **show snmp [mib | user | view | group | host | process-mib-time]**

【参数说明】 **mib** : 显示系统中支持的 snmp mib 信息。

user : 显示 snmp 用户信息。

view : 显示 snmp 视图信息。

group : 显示 snmp 用户组信息。

host : 显示用户配置的显示信息。

process-mib-time : 显示处理时间最长的 mib 节点。

【配置模式】 特权用户模式

【使用指导】 -

配置举例

SNMPv3 配置举例

【网络环境】

图 1-5



- 网络工作站(NMS)基于用户的认证加密模式对网络设备(Agent)进行管理。例如 :使用用户名 “user1”, 认证方式为 MD5, 认证密码为 123, 加密算法为 DES56, 加密密码为 321。
- 网络设备能够控制用户访问 MIB 对象的操作权限。例如 : 用户 “user1” 可以对 System (1.3.6.1.2.1.1) 节点下的 MIB 对象进行读操作, 其中只能对 SysContact (1.3.6.1.2.1.1.4.0) 节点下的 MIB 对象进行写操作。
- 网络设备能够主动向网管工作站发送验证加密的消息。

【配置方法】

- 第一步, 配置 MIB 视图和组。创建一个 MIB 视图 “view1”, 包含关联的 MIB 对象 (1.3.6.1.2.1.1); 再创建一个 MIB 视图 “view2”, 包含关联的 MIB 对象 (1.3.6.1.2.1.1.4.0)。创建一个组 “g1”, 选择版本号为 “v3”, 配置安全级别为认证加密模式 “priv”, 并可读视图 “view1”, 可写视图 “view2”。
- 第二步, 配置 SNMP 用户。创建用户名 “user1”, 属于组 “g1”, 选择版本号为 “v3”, 配置认证方式为 “md5”, 认证密码为 “123”, 加密方式为 “DES56”, 加密密码为 “321”。
- 第三步, 配置 SNMP 主机地址。配置主机地址为 192.168.3.2, 选择版本号为 “3”, 配置安全级别为认证加密模式 “priv”, 关联对应的用户名 “user1”。使能 Agent 主动向 NMS 发送 Trap 消息。
- 第四步, 配置 Agent 的 IP 地址。配置 Gi0/1 的接口地址为 192.168.3.1/24。

Agent

```

Ruijie(config)#snmp-server view view1 1.3.6.1.2.1.1 include
Ruijie(config)#snmp-server view view2 1.3.6.1.2.1.1.4.0 include
Ruijie(config)#snmp-server group g1 v3 priv read view1 write view2
Ruijie(config)#snmp-server user user1 g1 v3 auth md5 123 priv des56 321
Ruijie(config)#snmp-server host 192.168.3.2 traps version 3 priv user1
  
```

```
Ruijie(config)#snmp-server enable traps
Ruijie(config)#interface gigabitEthernet 0/1
Ruijie(config-if-gigabitEthernet 0/1)#ip address 192.168.3.1 255.255.255.0
Ruijie(config-if-gigabitEthernet 0/1)#exit
```

【检验方法】

- 第一步，通过 **show running-config** 命令查看设备的配置信息。
- 第二步，通过 **show snmp user** 命令查看 SNMP 用户。
- 第三步，通过 **show snmp view** 命令查看 SNMP 视图。
- 第四步，通过 **show snmp group** 命令查看 SNMP 组。
- 第五步，通过 **show snmp host** 命令查看用户配置的主机信息。
- 第六步，安装 MIB-Browser 查询。

Agent

```
Ruijie# show running-config
!
interface gigabitEthernet 0/1
 no ip proxy-arp
 ip address 192.168.3.1 255.255.255.0
!
snmp-server view view1 1.3.6.1.2.1.1 include
snmp-server view view2 1.3.6.1.2.1.1.4.0 include
snmp-server user user1 gl v3 encrypted auth md5 7EBD6A1287D3548E4E52CF8349CBC93D priv des56
D5CEC4884360373ABBF30AB170E42D03
snmp-server group gl v3 priv read view1 write view2
snmp-server host 192.168.3.2 traps version 3 priv user1
snmp-server enable traps

Ruijie# show snmp user
User name: user1
Engine ID: 800013110300d0f8221120
storage-type: permanent      active
Security level: auth priv
Auth protocol: MD5
Priv protocol: DES
Group-name: gl

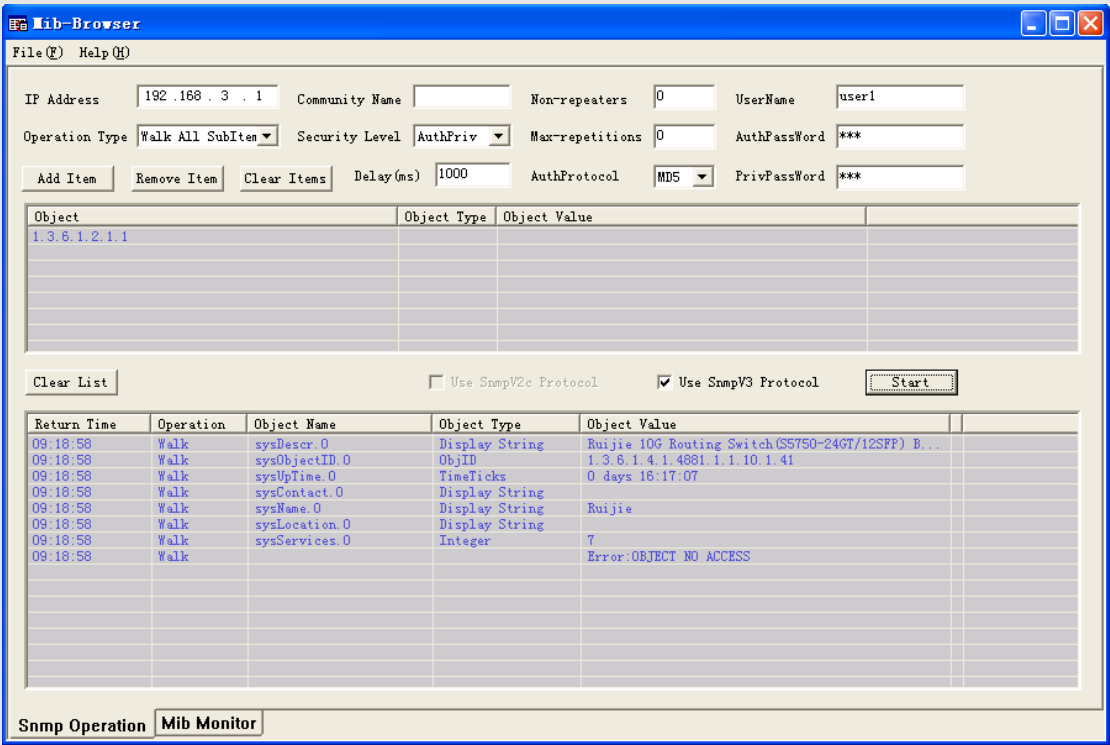
Ruijie#show snmp view
view1(include) 1.3.6.1.2.1.1
view2(include) 1.3.6.1.2.1.1.4.0
default(include) 1.3.6.1

Ruijie# show snmp group
groupname: gl
securityModel: v3
securityLevel:authPriv
readview: view1
```

```
writeview: view2
notifyview:

Ruijie#show snmp host
Notification host: 192.168.3.2
udp-port: 162
type: trap
user: user1
security model: v3 authPriv
```

安装MIB-Browser 在 IP Address 中输入设备的 IP 地址 :192.168.3.1 在UserName 中输入“user1” 在Security Level 中选择 “AuthPriv” ,在 AuthPassWord 中输入 “123” ,在 AuthProtocol 中选择 “MD5” ,在 PrivPassWord 中输入 “321”。点击 add item 按钮,选择要查询的 MIB 的具体管理单元,比如下图的 System。点击 Start 按钮,便开始对网络设备进行 MIB 的查询了,具体的查询结果见对话框的最下面的窗口：



常见错误

1.4.2 启用 Trap 功能

配置效果

使 Agent 主动向 NMS 发送 Trap 消息。

注意事项

-

配置方法

配置 snmp 主机地址

- 可选配置。
- 需要 Agent 主动发送消息时需要配置 NWS 的主机地址。

Agent 主动向 NMS 发送 Trap 消息

- 可选配置。
- 当需要 agent 主动向 NMS 发送 Trap 消息时，需在 agent 上配置此项。

打开接口发送 Link Trap 功能

- 可选配置。
- 当需要接口发送 link trap 功能时，需在 agent 上配置接口打开此项。

打开发送系统重启 Trap 功能

- 可选配置。
- 当希望 RGOS 系统在设备 **reload/reboot** 以前给 NMS 发送 Trap 消息通知系统重启时，需在 agent 上配置此项。

指定发送 Trap 消息的源地址

- 可选配置。
- 当希望固定使用一个本地 IP 地址作为 SNMP 的源地址以便于管理时，需在 agent 上配置此项。

发送 Trap 消息时携带私有字段

- 可选配置。
- 当需要 Trap 消息携带私有字段时，需在 agent 上配置此项。

检验方法

通过 **show snmp** 命令显示 SNMP 的状态信息。

通过 **show running-config** 命令查看设备的配置信息。

相关命令

配置 NMS 主机地址

【命令格式】 **snmp-server host** [oob] { *host-addr* | **ipv6** *ipv6-addr* | **domain** *domain-name* } [**vrf** *vrfname*] [**traps** | **informs**] [**version** { **1** | **2c** | **3** { **auth** | **noauth** | **priv** }] *community-string* [**udp-port** *port-num*] [*notification-type*]

【参数说明】 **oob**：将告警服务器指定为带外通信（一般指通过 MGMT 口发往告警服务器）

host-addr：SNMP 主机地址。

ipv6-addr：SNMP 主机地址（ipv6）。

domain-name：SNMP 主机域名

Vrfname：配置 vrf 转发表名称。

traps | **informs**：配置主机发送 trap 报文还是 inform 报文。


Version：选择 snmp 版本，V1、V2C、V3。

auth | **noauth** | **priv**：配置 V3 用户的安全级别。

community-string：团体字符串或用户名（V3 版本）。

port-num：配置 snmp 主机端口。

notification-type：主动发送的 Trap 类型，例如 snmp。

 如果没有指定 Trap 类型，则包括所有 Trap 类型。

【命令模式】 全局配置模式

【使用指导】 该命令与全局配置命令 **snmp-server enable traps** 一起使用，主动给 NMS 发送 Trap 消息。
可以配置多个不同的 SNMP 主机用于接收 Trap 消息，一个主机可以使用不同 Trap 类型组合，不同的端口，不同的 vrf 转发表，对于相同主机（相同端口、相同 vrf 配置），最后的一次配置会和前面的配置合并，即如要给相同主机发送不同 Trap 消息，可以分别配置不同 Trap 类型，最终这些配置会合并到一起。

配置 Agent 主动向 NMS 发送 Trap 消息

【命令格式】 **snmp-server enable traps** [*notification-type*]

【参数说明】 *notification-type*：启用对应事件的 Trap 通知，有以下类型：

snmp：启动 SNMP 事件的 TRAP 通知；

bgp：启动 BGP 事件的 TRAP 通知；

bridge：启动 BRIDGE 事件的 TRAP 通知；

isis：启动 ISIS 事件的 TRAP 通知；

mac-notification：启动 MAC 事件的 TRAP 通知；

ospf：启动 OSPF 事件的 TRAP 通知；

urpf：启动 URPF 事件的 TRAP 通知；

vrrp：启动 VRRP 事件的 TRAP 通知；

web-auth：启动 WEB 认证事件的 TRAP 通知。

【命令模式】 全局配置模式

【使用指导】 该命令必须与全局配置命令 **snmp-server host** 一起使用，才能发送 Trap 消息。

打开接口发送 Link Trap 功能

【命令格式】 **snmp trap link-status**

【参数说明】 -

【配置模式】 接口配置模式

- 【使用指导】 对于接口（以太网接口、Ap 接口、SVI 接口），当功能打开时，如果接口发生 Link 状态变化，SNMP 将发出 Link Trap，反之则不发。

✎ 打开发送系统重启 Trap 功能

- 【命令格式】 **snmp-server system-shutdown**
- 【参数说明】 -
- 【配置模式】 全局配置模式
- 【使用指导】 打开 SNMP 系统重启通知功能，会在设备 **reload/reboot** 以前给 NMS 发送 Trap 消息通知系统重启。

✎ 指定发送 Trap 消息的源地址

- 【命令格式】 **snmp-server trap-source interface**
- 【参数说明】 *interface*：用于作为 SNMP 源地址的接口。
- 【配置模式】 全局配置模式
- 【使用指导】 缺省情况下，SNMP 报文从哪个接口出去，就使用哪个接口的 IP 地址作为源地址，为了便于管理和识别，可以使用该命令固定使用一个本地 IP 地址作为 SNMP 的源地址。

✎ 配置发送 Trap 消息时携带私有字段

- 【命令格式】 **snmp-server trap-format private**
- 【参数说明】 -
- 【配置模式】 全局配置模式
- 【使用指导】 使用该命令可配置发送 Trap 消息携带私有格式字段，包含的字段目前支持的有告警发生时间，各个字段的具体数据类型和数据范围可参见 RUIJIE-TRAP-FORMAT-MIB.mib 文件说明。

配置举例

✎ 配置启用 trap 功能

【网络环境】

图 1-6



- 网管工作站（NMS）基于共同体认证模式对网络设备（Agent）进行管理，网络设备能够主动向网管工作站发送消息。

- 【配置方法】
- 第一步，配置 Agent 主动向 NMS 发送消息。配置 SNMP 主机地址为 192.168.3.2，消息格式为 Version 2c，认证名为 “user1”。使能 Agent 主动发送 Trap 消息。
 - 第二步，配置 Agent 的 IP 地址。配置 Gi 0/1 的接口地址为 192.168.3.1/24。

Agent

```
Ruijie(config)#snmp-server host 192.168.3.2 traps version 2c user1
Ruijie(config)#snmp-server enable traps
Ruijie(config)#interface gigabitEthernet 0/1
```

```
Ruijie(config-if-gigabitEthernet 0/1)#ip address 192.168.3.1 255.255.255.0
Ruijie(config-if-gigabitEthernet 0/1)#exit
```

- 【检验方法】
- 通过 **show running-config** 命令查看设备的配置信息。
 - 通过 **show snmp** 命令显示 SNMP 的状态信息。

Agent

```
Ruijie# show running-config
ip access-list standard a1
  10 permit host 192.168.3.2
interface gigabitEthernet 0/1
  no ip proxy-arp
  ip address 192.168.3.1 255.255.255.0
snmp-server view v1 1.3.6.1.2.1.1 include
snmp-server location fuzhou
snmp-server host 192.168.3.2 traps version 2c user1
snmp-server enable traps
snmp-server contact ruijie.com.cn
snmp-server community user1 view v1 rw a1
snmp-server chassis-id 1234567890

Ruijie#show snmp
Chassis: 1234567890
0 SNMP packets input
    0 Bad SNMP version errors
    0 Unknown community name
    0 Illegal operation for community name supplied
    0 Encoding errors
    0 Number of requested variables
    0 Number of altered variables
    0 Get-request PDUs
    0 Get-next PDUs
    0 Set-request PDUs
0 SNMP packets output
    0 Too big errors (Maximum packet size 1472)
    0 No such name errors
    0 Bad values errors
    0 General errors
    0 Response PDUs
    0 Trap PDUs
SNMP global trap: enabled
SNMP logging: disabled
SNMP agent: enabled
```

常见错误

-

1.4.3 屏蔽 Agent 功能

配置效果

在不需要 Agent 服务的时候，屏蔽 Agent 功能。

注意事项

- 执行 **no snmp-server** 命令，可以在不需要代理服务的时候，屏蔽 SNMP 代理功能。
- 不同于屏蔽命令，执行 **no enable service snmp-agent** 命令，会直接关闭 snmp 所有服务（即 snmp 代理功能被禁用了，不接收报文、不发送响应报文及 trap），不会屏蔽代理的配置信息。

配置方法

📌 配置屏蔽设备 SNMP 代理

- 可选配置。
- 需要屏蔽所有 SNMP 代理服务配置时，可选用此项配置。

📌 配置关闭设备 SNMP 代理

- 可选配置。
- 需要直接关闭所有服务时，应选用此配置项。

检验方法

通过 **show services** 命令查看 snmp 服务的开关状态信息。

通过 **show snmp** 命令显示 SNMP 的状态信息。

通过 **show running-config** 命令查看设备的配置信息。

相关命令

📌 配置屏蔽设备 SNMP 代理功能

【命令格式】 **no snmp-server**

【参数说明】 -

- 【命令模式】全局配置模式
- 【使用指导】SNMP 代理功能服务默认关闭，在设置 SNMP 代理参数（例如 NMS 主机地址、认证名和访问权限等）时，会自动打开 SNMP 代理服务，服务开关命令 **enable service snmp-agent** 也必须同时打开，SNMP 代理服务才能生效，但只要关闭了其中的一个，SNMP 代理服务将不会生效。使用 **no snmp-server** 命令可以关闭设备支持的所有版本 SNMP 的代理服务。
使用该命令的同时，将屏蔽所有 SNMP 代理服务配置（即使用 **show running-config** 命令查看时不会显示配置，重新开启 SNMP 代理服务可以恢复），而 **enable service snmp-agent** 命令则不会屏蔽 SNMP 代理配置。

配置关闭设备 SNMP 代理功能

- 【命令格式】no enable service snmp-agent
- 【参数说明】-
- 【配置模式】全局配置模式
- 【使用指导】关闭 SNMP 服务开关，但不会屏蔽 SNMP 代理参数。

配置举例

配置启用 snmp 服务功能

【网络环境】

图 1-7



通过设置 snmp 服务开关，以及设置 snmp 代理服务器，使得网管工作站（NMS）能通过 snmp 访问设备。

- 【配置方法】
- 配置启用 snmp 服务。
 - 配置 snmp 代理服务器的参数，使服务生效。

A gent `Ruijie(config)#enable service snmp-agent`

- 【检验方法】
- 通过 **show services** 命令查看 snmp 服务的开关状态信息。

Agent `Ruijie#show service`
`web-server : disabled`
`web-server(https): disabled`
`snmp-agent : enabled`
`ssh-server : disabled`
`telnet-server : enabled`

常见错误

-

1.4.4 设置 SNMP 控制参数

配置效果

对 SNMP 的 Agent 的基本参数进行配置，包括设备的联系方式、设备位置、序列号、发送 Trap 消息的参数等，NMS 通过访问设备的这些参数，便可以得知设备的联系人，设备所在的物理位置等信息。

注意事项

-

配置方法

✎ 配置系统的联系方式

- 可选配置。
- 当需要修改系统的联系方式时，需在 agent 上配置此项。

✎ 配置系统位置

- 可选配置。
- 当需要修改系统的系统位置时，需在 agent 上配置此项。

✎ 配置系统序列码

- 可选配置。
- 当需要修改系统的序列码时，需在 agent 上配置此项。

✎ 配置设备的网元信息

- 可选配置。
- 当需要修改网元编码信息时，需在 agent 上配置此项。

✎ 配置 SNMP 代理最大数据报文长度

- 可选配置。
- 当需要修改 SNMP 代理最大数据报文长度时，需在 agent 上配置此项。

✎ 配置 SNMP 服务 UDP 端口号

- 可选配置。
- 当需要修改 SNMP 服务的 UDP 端口号时，需在 agent 上配置此项。

✎ 配置 Trap 消息报文的队列长度

- 可选配置。
- 当希望通过调整消息队列大小来控制消息发送速度时，需在 agent 上配置此项。

配置发送 Trap 消息的时间间隔

- 可选配置。
- 当需要修改发送 Trap 消息的时间间隔时，需在 agent 上配置此项。

配置 SNMP 流控

- 可选配置。
- 如果 SNMP 的请求报文太多导致 SNMP 任务的 CPU 占用比较高，可以配置 SNMP 流控，限制 SNMP 任务每秒处理的请求报文个数，从而控制 SNMP 任务的 CPU 占用情况。

检验方法

通过 **show snmp** 命令显示 SNMP 的状态信息。

通过 **show running-config** 命令查看设备的配置信息。

相关命令

配置系统的联系方式

- 【命令格式】 **snmp-server contact text**
- 【参数说明】 *text*：描述系统联系方式的字符串。
- 【命令模式】 全局配置模式
- 【使用指导】

配置系统位置

- 【命令格式】 **snmp-server location text**
- 【参数说明】 *text*：描述系统信息的字符串。
- 【配置模式】 全局配置模式
- 【使用指导】

配置系统序列码

- 【命令格式】 **snmp-server chassis-id text**
- 【参数说明】 *text*：系统序列号的文本，可以是数字或字符。
- 【配置模式】 全局配置模式
- 【使用指导】 SNMP 系统序列号一般使用机器的序列号，以便对设备进行识别。

配置设备的网元信息

- 【命令格式】 **snmp-server net-id text**

- 【参数说明】 *text*：设置设备网元编码 *text*，*text* 是长度为 1~255 的字符串，区分大小写，可包含空格。
- 【配置模式】 全局模式
- 【使用指导】 配置设备网元编码信息。

配置 SNMP 代理最大数据报文长度

- 【命令格式】 **snmp-server packetsize** *byte-count*
- 【参数说明】 *byte-count*：数据包大小，从 484 字节到 17876 字节。
- 【配置模式】 全局模式
- 【使用指导】

配置 SNMP 服务 UDP 端口号

- 【命令格式】 **snmp-server udp-port** *port-num*
- 【参数说明】 *port-num*：指定 SNMP 服务的 UDP 端口号，即接收 SNMP 报文的协议端口号。
- 【配置模式】 全局模式
- 【使用指导】 指定接收 SNMP 报文的协议端口号。

配置 Trap 消息报文的队列长度

- 【命令格式】 **snmp-server queue-length** *length*
- 【参数说明】 *length*：队列长度，大小从 1 到 1000。
- 【配置模式】 全局配置模式
- 【使用指导】 通过调整消息队列大小来控制消息发送速度。

配置发送 Trap 消息的时间间隔

- 【命令格式】 **snmp-server trap-timeout** *seconds*
- 【参数说明】 *seconds*：间隔时间，单位为秒，取值范围：1 – 1000。
- 【配置模式】 全局配置模式
- 【使用指导】 通过调整发送消息的时间间隔来控制消息发送速度。

配置 SNMP 流控

- 【命令格式】 **snmp-server flow-control pps** [*count*]
- 【参数说明】 *count*：每秒处理的 SNMP 请求报文数量，范围<50-65535>。
- 【命令模式】 全局配置模式
- 【使用指导】 如果 SNMP 的请求报文太多导致 SNMP 任务的 CPU 占用比较高，可以配置 SNMP 流控，限制 SNMP 任务每秒处理的请求报文个数，从而控制 SNMP 任务的 CPU 占用情况。

配置 SNMP 版本

- 【命令格式】 **snmp-server enable version** [*v1* | *v2c* | *v3*]
- 【参数说明】 *v1*、*v2c*、*v3* 表示 SNMP 的三个协议版本
- 【命令模式】 全局配置模式
- 【使用指导】 如果用户场景中，仅使用指定的协议版本进行 MIB 交互管理，可以将其他版本功能关闭

配置举例

设置 SNMP 的控制参数

【网络环境】

图 1-8



- 网管工作站（NMS）基于共同体认证模式对网络设备（Agent）进行管理，网管工作站能够获取设备的基本系统信息，如系统的联系方式、位置、序列码。

【配置方法】

- 第一步，配置 SNMP 代理参数。配置系统所处的位置、联系方式、序列码。
- 第二步，配置 Agent 的 IP 地址。配置 Gi 0/1 的接口地址为 192.168.3.1/24。

Agent

```
Ruijie(config)#snmp-server location fuzhou
Ruijie(config)#snmp-server contact ruijie.com.cn
Ruijie(config)#snmp-server chassis-id 1234567890
Ruijie(config)#interface gigabitEthernet 0/1
Ruijie(config-if-gigabitEthernet 0/1)#ip address 192.168.3.1 255.255.255.0
Ruijie(config-if-gigabitEthernet 0/1)#exit
```

【检验方法】

- 第一步，查看设备的配置信息。
- 第二步，查看 SNMP 视图和组的信息。

Agent

```
Ruijie# show running-config
ip access-list standard a1
 10 permit host 192.168.3.2
interface gigabitEthernet 0/1
 no ip proxy-arp
 ip address 192.168.3.1 255.255.255.0
snmp-server view v1 1.3.6.1.2.1.1 include
snmp-server location fuzhou
snmp-server host 192.168.3.2 traps version 2c user1
snmp-server enable traps
snmp-server contact ruijie.com.cn
snmp-server community user1 view v1 rw a1
snmp-server chassis-id 1234567890

Ruijie#show snmp view
v1(include) 1.3.6.1.2.1.1
default(include) 1.3.6.1
```

```
Ruijie#show snmp group
groupname: user1
securityModel: v1
securityLevel:noAuthNoPriv
readview: v1
writeview: v1
notifyview:
groupname: user1
securityModel: v2c
securityLevel:noAuthNoPriv
readview: v1
writeview: v1
notifyview:
```

常见错误

-

1.5 监视与维护

清除各类信息

清除 SNMP 连续认证失败后被锁定的源 IP 地址表	<code>clear snmp locked-ip [ipv4 ipv4-address ipv6 ipv6-address]</code>
-----------------------------	---

查看运行情况

作用	命令
显示 SNMP 的状态信息	<code>show snmp [mib user view group host]</code>

2 NETCONF

2.1 概述

NETCONF 协议(Network Configuration Protocol 即 IETF 的 NETCONF 工作组在 2003 年提出的一个全新的基于 XML 的网络配置协议,可以对设备进行配置、参数检索以及监控管理等。协议通信模式是采用 C/S 模式,设备上运行的是协议的服务器程序,而用户运行的是协议的客户端程序。协议报文格式是 XML 格式,包括所有配置数据和协议消息都是 XML 格式。协议分四层:内容层、操作层、RPC 层、传输层。其中,内容层是被管理的数据对象集合,设备的配置数据就是这一层;操作层是 RPC 中应用的基本原语操作集,如 get、get-config、edit-config、delete-config 等;RPC 层提供了一个简单的、传输协议无关的机制,包含一些错误反馈消息元素的规定;传输层是提供安全传输通道,协议支持 SSH、SOAP、BEEP (SSH 是强制的)。下文仅介绍 NETCONF 的相关内容。

协议规范

- RFC4741: NETCONF Configuration Protocol、
- RFC4742: Using the NETCONF Configuration Protocol over Secure Shell (SSH)
- RFC4743: Using NETCONF over the Simple Object Access Protocol (SOAP)
- RFC4744: Using the NETCONF Protocol over the Blocks Extensible Exchange Protocol (BEEP)
- RFC5277: NETCONF Event Notifications
- RFC5381: Experience of Implementing NETCONF over SOAP
- RFC5539: NETCONF Over Transport Layer Security (TLS)
- RFC5717: Partial Lock RPC for NETCONF
- RFC6022: NETCONF Monitoring Schema
- RFC6241: Network Configuration Protocol
- RFC6242: Using the Network Configuration Protocol over Secure Shell
- RFC6243: With-defaults capability for NETCONF
- RFC6470: NETCONF Notification Events
- RFC6536: NETCONF Access Control Model (NACM)

其中 RFC4741 和 RFC4742 分别被 RFC6241 和 RFC6242 取代。

2.2 典型应用

典型应用	场景描述
NETCONF 网络设备管理	用户通过网管软件(NETCONF 客户端)向设备发送 NETCONF 协议配置报文(XML),

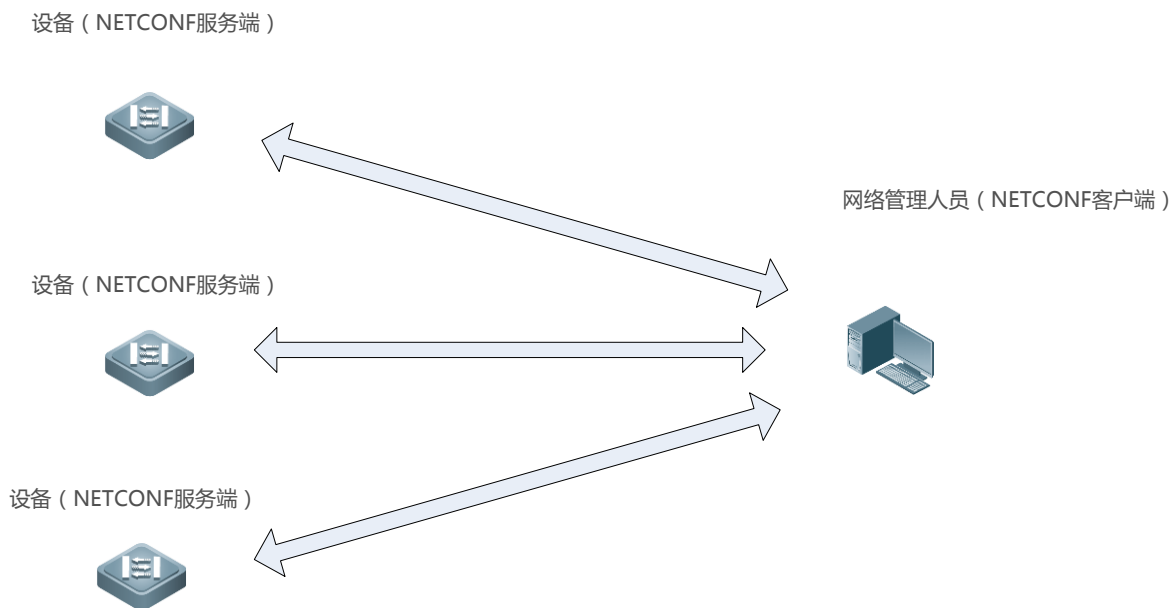
以实现对设备的配置管理。

2.2.1 NETCONF 网络设备管理

应用场景

以下图为例，用户通过 NETCONF 网络管理软件，来对网络设备进行管理和监控。

图 2-1



功能部署

网络管理站和被管理的网络设备通过网络连接，管理软件和设备都需要运行 SSH 协议；用户在网络管理站上，通过 NETCONF 网络管理软件，访问网络设备上的配置数据库和状态数据库，以及接收来自网络设备主动发出的事件通告消息，来对网络设备进行管理和监控。

2.3 功能详解

基本概念

常用术语

- RPC : Remote Procedure Call (远程过程调用)。

- DM : Data Model (数据模型)。

协议结构

NETCONF 的协议结构如下图所示：

图 2-2 NETCONF 的协议结构



会话连接

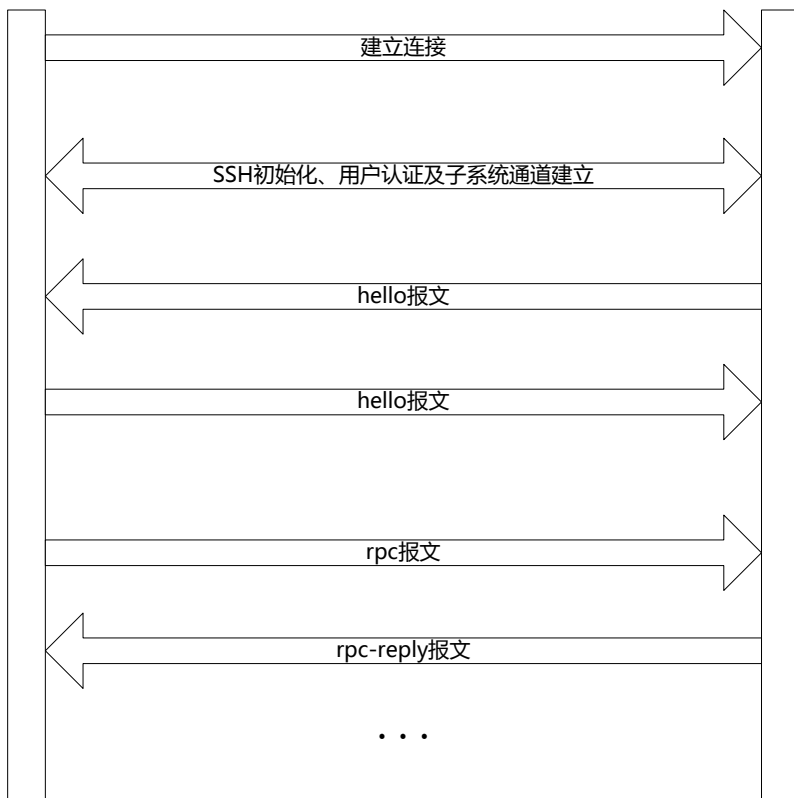
- NETCONF over SSH

服务端在端口 830 监听，使用前需要建立 SSH 通道，经过一系列的传输算法协商（这其中包含密钥协商、压缩算法、哈希算法、加密算法、签名算法等）和用户认证后建立 SSH 通道（SSH 子系统，命名为 netconf）。

图 2-3 NETCONF 会话交互报文示意图

NETCONF客户端

NETCONF服务端（设备端）



能力集交换

在通道建立之后，需要进行能力集交换，双方提供各自实现的能力集，对自己不理解的或者没有实现的能力进行忽略。但是每一方至少都必须支持协议基本能力（urn:ietf:params:netconf:base:1.1）。如果兼容旧协议版本需要支持旧协议基本能力（urn:ietf:params:netconf:base:1.0）。

协议操作

NETCONF 操作层定义了 9 种基础的操作方法，分类为，get、get-config 是取值操作，edit-config、copy-config、delete-config 是配置操作，lock 与 unlock 是对设备临界资源（配置文件等）并发操作时的锁保护，close-session、kill-session 是结束会话操作。

- get：获取设备状态和配置数据；
- get-config：根据操作内容的过滤节点获取相应的配置数据；
- edit-config：根据提供数据模型定义以及操作属性进行设备配置，其中一个比较重要的属性是 operation 包括 merge、replace、create、delete、remove；相关参数 default-operation 默认值是 merge；
- copy-config：配置文件拷贝，如，将候选配置拷贝至配置文件，启机配置拷贝至运行配置，运行配置写入启机配置等，这些需要目标文件支持可写的能力；
- delete-config：删除设备配置文件，但不允许删除设备的运行文件；

- lock：对配置数据文件进行锁保护，允许当前客户端进行访问（或者修改）同时其他客户端或者非 NETCONF 客户端（如 SNMP 或者 CLI）无法访问（或者修改）；
- unlock：对配置数据文件进行解锁操作；
- close-session：关闭当前会话，包括资源、锁的释放以及连接断开等，这些操作必须保证当前业务处理完毕，且不再处理新的请求；
- kill-session：强制关闭会话（但是不允许关闭当前会话），包括资源、锁的释放以及连接断开等，如果有当前业务在处理，必须停止，未完成的业务必须回滚至业务处理开始之前。


功能特性

功能特性	作用
能力集交换	NETCONF 服务端和客户端相互发送能力集，在协议版本上，服务端（设备端）支持协议版本 1.0 和 1.1
获取数据操作	客户端获取设备配置数据或者状态数据
获取配置操作	客户端获取设备配置数据
编辑配置操作	客户端修订设备配置数据
拷贝配置操作	客户端拷贝设备上的某个配置文件到另一个配置文件
删除配置操作	客户端删除设备上的整个配置文件
关闭会话	客户端主动关闭当前与设备交互的 NETCONF 会话
配置上锁	客户端给设备整个配置文件上锁
配置解锁	客户端给设备整个配置文件解锁

2.3.1 能力集交换

客户端与服务端建立连接之后，双方立即交互各自的能力集，在双方都有支持相同的 NETCONF 协议版本之后，才可以进行后续的数据操作。发送报文格式如下：

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>
      urn:ietf:params:netconf:base:1.1
    </capability>
    <capability>
      能力集 1
    </capability>
    <capability>
      能力集 2
    </capability>
  </capabilities>
  <session-id>会话 ID</session-id>
</hello>
```


 客户端发给服务端的能力交互报文，不得带有会话 ID 节点（<session-id>）。

能力集交互举例，服务端：

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>urn:ietf:params:netconf:base:1.0</capability>
    <capability>urn:ietf:params:netconf:base:1.1</capability>
    <capability>urn:ietf:params:netconf:capability:writable-running:1.0</capability>
    <capability>urn:ietf:params:xml:ns:yang:ietf-yang-types?module=ietf-yang-types&revision=2013-07-15</capability>
    <capability>urn:rg:params:xml:ns:yang:rg-tacacs?module=rg-tacacs&revision=2016-10-25</capability>
    <capability>urn:rg:params:xml:ns:yang:rg-interfaces?module=rg-interfaces&revision=2016-10-25</capability>
    <capability>urn:ietf:params:xml:ns:yang:ietf-inet-types?module=ietf-inet-types&revision=2010-09-24</capability>
    <capability>urn:rg:params:xml:ns:yang:rg-openflow?module=rg-openflow&revision=2016-09-26</capability>
  </capabilities>
  <session-id>28</session-id>
</hello>
```

能力集交互举例，客户端：

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>urn:ietf:params:netconf:base:1.0</capability>
  </capabilities>
</hello>
```

2.3.2 获取数据操作

获取设备配置或者状态数据。

客户端发送报文格式如下：

```
<?xml version="1.0" encoding="utf-8"?>
<rpc message-id="xxx " xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-config>
    <filter type="subtree">
      配置数据（或者状态数据）过滤规则
    </filter>
  </get-config>
</rpc>
```

服务端应答报文格式如下：

```
<?xml version="1.0" encoding="utf-8"?>
<rpc-reply message-id="xxx " xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

获取到的配置数据（或者状态数据）

```
</data>
```

```
</rpc-reply>
```

如果设备上状态数据的所有子集都无法匹配过滤规则，则会应答空的 data 节点，如下：

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<rpc-reply message-id="消息 ID " xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
  <data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"/>
```

```
</rpc-reply>
```

2.3.3 获取配置操作

获取设备配置数据，该操作通过各种子树过滤规则获取相应的配置数据子集，但是不能获取设备状态数据。

客户端发送报文格式如下：

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<rpc message-id="xxx" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
  <get-config>
```

```
    <source>
```

```
      <running/>
```

```
    </source>
```

```
    <filter type="subtree">
```

```
      协议过滤规则
```

```
    </filter>
```

```
  </get-config>
```

```
</rpc>
```

服务端应答报文格式如下：

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<rpc-reply message-id="xxx" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
  <data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    获取到的配置数据
```

```
  </data>
```

```
</rpc-reply>
```

如果设备上配置数据的所有子集都无法匹配过滤规则，则会应答空的 data 节点，如下：

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<rpc-reply message-id="xxx " xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
  <data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"/>
```

```
</rpc-reply>
```

2.3.4 编辑配置操作

根据提供的数据模型定义以及操作属性进行配置编辑。配置编辑报文中，包含 5 个操作属性，在下发的 XML 报文中配置节点的 operation 属性描述中给出，这 5 个操作属性分别是：

merge：将 edit-config 报文中包含该属性配置数据合并到指定的设备配置文件中（或者数据库），如果配置数据不存在，则直接根据下发内容创建该配置数据；

replace：将 edit-config 报文中包含该属性配置数据替换指定的设备配置文件中（或者数据库）的相应配置数据节点，如果该配置数据不存在则直接根据下发内容创建该配置；目前锐捷设备对这个操作暂不支持，如果有下发该属性操作还是按 merge 操作处理；

create：在指定配置数据文件中（或者数据库）创建 edit-config 报文中包含该属性的配置数据；如果配置数据不存在，则直接根据下发内容创建该配置数据；如果配置数据已经存在，则会应答 rpc-error 报文，error-tag 指示 data-exists；

delete：在指定配置数据文件中（或者数据库）删除 edit-config 报文中包含该属性的配置数据；如果配置数据存在，则直接删除相应的配置；如果配置数据不存在，则会应答 rpc-error 报文，error-tag 指示 data-missing；

remove：在指定配置数据文件中（或者数据库）移除 edit-config 报文中包含该属性的配置数据；如果配置数据存在，则直接移除相应的配置；如果配置数据不存在，则忽略该操作返回 ok。

客户端发送报文格式如下：

```
<?xml version="1.0" encoding="utf-8"?>
<rpc message-id="xxx" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target> <running/> </target>
    <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">
      配置数据
    </config>
  </edit-config>
</rpc>
```

报文中没有携带 error-option 节点，默认是该节点值为 stop-on-error，即一旦遇到哪个节点配置出错则立刻停止同一个报文中剩余的后续配置并返回错误(rpc-error)；报文中没携带 test-option 节点，默认该节点的值是 test-then-set；报文中没携带 default-operation 节点，默认该节点的值是 merge 操作。

服务端应答报文格式如下：

```
<?xml version="1.0" encoding="utf-8"?>
<rpc-reply message-id="消息 ID" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

报文有携带 error-option，一般如下格式：

```
<?xml version="1.0" encoding="utf-8"?>
<rpc message-id="xxx" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
```

```

<target> <running/> </target>
<error-option>配置出错时的行为选项</error-option>
<config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">
    配置数据
</config>
</edit-config>
</rpc>

```

error-option 节点是枚举类型的节点，设备可以支持的值有：

stop-on-error：edit-config 操作时，遇到第一个出错地方，立刻停止当前的 edit-config 操作，当前配置报文前面配置的数据都已经生效（出错之前的配置），这个值是 error-option 的缺省值；

continue-on-error：edit-config 操作时，遇到配置出错会记录当前的错点并继续处理剩余的配置，但是最后返回错误信息（即出现任何配置错误，最终的应答报文还是 rpc-error）；

2.3.5 拷贝配置操作

将启机配置同步至运行配置。

客户端发送格式如下：

```

<?xml version="1.0" encoding="utf-8"?>
<rpc message-id="xxx" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <copy-config>
        <target>
            <startup/>
        </target>
        <source>
            <running/>
        </source>
    </copy-config>
</rpc>

```

服务端应答报文格式如下：

```

<?xml version="1.0" encoding="utf-8"?>
<rpc-reply message-id="xxx" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>

```

2.3.6 删除配置操作

将设备的启配置删除，运行配置是无法删除。

客户端发送格式如下：

```
<?xml version="1.0" encoding="utf-8"?>
<rpc message-id="xxx" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <delete-config>
    <target>
      <startup/>
    </target>
  </delete-config>
</rpc>
```

服务端应答报文格式如下：

```
<?xml version="1.0" encoding="utf-8"?>
<rpc-reply message-id="xxx" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

2.3.7 关闭会话

关闭当前会话，会释放资源和锁，断开连接。

客户端发送格式如下：

```
<?xml version="1.0" encoding="utf-8"?>
<rpc message-id="xxx" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <close-session/>
</rpc>
```

服务端应答报文格式如下：

```
<?xml version="1.0" encoding="utf-8"?>
<rpc-reply message-id="xxx" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

2.3.8 配置上锁

RFC6241 说明 lock 是用来锁定配置数据库（配置文件），防止多个源（如，CLI、SNMP 以及多个 NETCONF 会话并发等）并发针对的设备的配置文件进行修订，导致引入其他无关的配置修订。设备当前针对这个操作做一些阉割，只能防止多个 NETCONF 会话并发修订（运行配置），保证配置数据修订安全。

客户端发送格式如下：

```
<?xml version="1.0" encoding="utf-8"?>
<rpc message-id="xxx" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <lock>
    <target>
```

```
<running/>
</target>
</lock>
</rpc>
```

服务端应答报文格式如下：

```
<?xml version="1.0" encoding="utf-8"?>
<rpc-reply message-id="xxx " xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

2.3.9 配置解锁

unlock 是用来解锁配置数据库（配置文件，设备这里指的是运行配置），与 lock 操作是成对操作。

客户端发送格式如下：

```
<?xml version="1.0" encoding="utf-8"?>
<rpc message-id="xxx " xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <unlock>
    <target>
      <running/>
    </target>
  </unlock>
</rpc>
```

服务端应答报文格式如下：

```
<?xml version="1.0" encoding="utf-8"?>
<rpc-reply message-id="xxx " xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

2.4 配置详解

使用 NETCONF 前需要配置 SSH 通道认证相关参数，SSH 配置参见 SSH-SCG。

配置项	配置建议 & 相关命令	
配置 NETCONF 的 candidate 能力	 可选配置。NETCONF 服务端与客户端在能力交互时，服务端可配置反馈 candidate 能力于客户端。	
	netconf capability candidate	打开 NETCONF 的 candidate 、 confirmed-commit 能力。

配置 NETCONF 的 rollback-on-error 能力	 可选配置。NETCONF 服务端与客户端在能力交互时，服务端可配置反馈 rollback-on-error 能力于客户端。	
	netconf capability rollback	打开 NETCONF 的 rollback-on-error 能力。
配置 NETCONF 的 validate 能力	 可选配置。NETCONF 服务端与客户端在能力交互时，服务端可配置反馈 validate 能力于客户端。	
	netconf capability validate	打开 NETCONF 的 validate 能力。
配置 YANG 模块 feature 功能	 可选配置。NETCONF 服务端与客户端在能力交互时，服务端可配置不反馈 Feature 属性于客户端。	
	netconf feature-disable	关闭 NETCONF 的 feature 功能。
配置 netconf 会话查询返回最大节点数量	 可选配置。NETCONF 查询配置时，拒绝超过返回超过配置上限的节点数量。	
	netconf filter nodes-limit	配置 netconf 会话查询配置可返回的最大节点数量。
配置 netconf 会话最大连接数	 可选配置。NETCONF 服务端与客户端连接时，拒绝超过连接上限的会话连接成功。	
	netconf max-sessions	配置 netconf 会话最大连接数。
配置 YANG 模块多版本通告	 可选配置。NETCONF 服务端与客户端在能力交互时，服务端需要将所有支持的 yang 模块的所有版本通告给客户端。	
	netconf yang multi-revision	配置 NETCONF YANG 模块多版本通告功能。
配置 NETCONF 优先支持 openconfig YANG 模型	 可选配置。NETCONF 服务端配置优先支持 openconfig YANG 模型。	
	netconf yang suite openconfig	配置 NETCONF 优先支持 openconfig YANG 模型。

2.4.1 配置 NETCONF 的 candidate 能力

配置效果

在与 NETCONF 客户端能力交互时 (Hello 报文 , capabilities 报文) , 返回 candidate、**confirmed-commit** 能力。

注意事项

-

配置方法

打开 netconf 的 candidate、confirmed-commit 能力

【命令格式】 **netconf capability candidate**

【参数说明】 -

【缺省配置】 打开 netconf 的 **candidate**、**confirmed-commit** 能力

【命令模式】 全局配置模式

【使用指导】 -

检验方法

-

配置举例

打开 netconf 的 candidate、confirmed-commit 能力

【网络环境】

图 2-5

NETCONF服务端

NETCONF客户端



【配置方法】 ● 打开 netconf 的 **candidate**、**confirmed-commit** 能力

NETCONF

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# netconf capability candidate
Ruijie(config)#
```

【检验方法】 -

常见错误

-

2.4.2 配置 NETCONF 的 rollback 能力

配置效果

在与 NETCONF 客户端能力交互时 (Hello 报文 , capabilities 报文), 返回 rollback 能力。

注意事项

-

配置方法

打开 netconf 的 rollback 能力

【命令格式】 **netconf capability rollback**

【参数说明】 -

【缺省配置】 打开 netconf 的 **rollback** 能力

【命令模式】 全局配置模式

【使用指导】 -

检验方法

-

配置举例

打开 netconf 的 rollback 能力

【网络环境】

图 2-5

NETCONF服务端

NETCONF客户端



【配置方法】 ● 打开 netconf 的 **rollback** 能力

NETCONF

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# netconf capability rollback
Ruijie(config)#
```

【检验方法】 -

常见错误

-

2.4.3 配置 NETCONF 的 validate 能力

配置效果

在与 NETCONF 客户端能力交互时 (Hello 报文 , capabilities 报文), 返回 validate 能力。

注意事项

-

配置方法

打开 netconf 的 validate 能力

【命令格式】 **netconf capability validate**

【参数说明】 -

【缺省配置】 打开 netconf 的 **validate** 能力

【命令模式】 全局配置模式

【使用指导】 -

检验方法

-

配置举例

打开 netconf 的 validate 能力

【网络环境】

图 2-5

NETCONF服务端

NETCONF客户端



【配置方法】 ● 打开 netconf 的 validate 能力

NETCONF

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# netconf capability validate
Ruijie(config)#
```

【检验方法】 -

常见错误

-

2.4.4 配置 YANG 模块 feature 功能

配置效果

在与 NETCONF 客户端能力交互时（Hello 报文，capabilities 报文），不返回 yang 模块的 feature 属性。

注意事项

-

配置方法

关闭 netconf 的 feature 属性功能

【命令格式】 **netconf feature-disable**

【参数说明】 -

【缺省配置】 关闭 netconf 的 feature 属性功能

【命令模式】 全局配置模式

【使用指导】 -

检验方法

-

配置举例

关闭 netconf 的 feature 属性功能

【网络环境】

图 2-5

NETCONF服务端

NETCONF客户端



【配置方法】 ● 配置 YANG 模块多版本通告

NETCONF Ruijie# configure terminal

```
Enter configuration commands, one per line. End with CNTL/Z.  
Ruijie(config)# netconf feature-disable  
Ruijie(config)#
```

【检验方法】 -

常见错误

-

2.4.5 配置 netconf 会话查询配置可返回的最大节点数量

配置效果

在 NETCONF 客户端查询时，只允许返回不超过配置上限的 XML 节点数量，超过则返回错误。

注意事项

该配置只影响 target 为 running 的查询结果。

配置方法

▾ 配置 netconf 会话查询配置可返回的最大节点数量

【命令格式】 **netconf filter nodes-limit** *num*

【参数说明】 *num*: 指定最大返回节点数，默认是 100000，取值范围 100-2147483647

【缺省配置】 默认上限数量为 100000 个

【命令模式】 全局配置模式

【使用指导】 -

检验方法

-

配置举例

▾ 配置 netconf 会话查询配置可返回的最大节点数量

【网络环境】

图 2-5

NETCONF服务端

NETCONF客户端



【配置方法】

- 配置 netconf 会话查询配置可返回的最大节点数量

NETCONF

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# netconf filter nodes-limit 101
Ruijie(config)#
```

【检验方法】

```
Ruijie (config)# show running-config | in filter nodes-limit
netconf filter nodes-limit 101
```

常见错误

-

2.4.6 配置 netconf 会话最大连接数

配置效果

在 NETCONF 客户端连接时，只允许指定数量的会话连接设备。

注意事项

-

配置方法

🔗 配置 netconf 会话最大连接数

- 【命令格式】 **netconf max-sessions num**
- 【参数说明】 *num*: 指定最大连接数，默认是 1，取值范围 1-36
- 【缺省配置】 默认连接上限数量为 1 个
- 【命令模式】 全局配置模式
- 【使用指导】 -

检验方法

配置举例

配置 netconf 会话最大连接数

【网络环境】

图 2-5

NETCONF服务端

NETCONF客户端



【配置方法】

- 配置 netconf 会话最大连接数

NETCONF

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# netconf max-sessions 15
Ruijie(config)#
```

【检验方法】

```
Ruijie (config)#show running-config | in max-sessions
netconf max-sessions 15
```

常见错误

2.4.7 配置 YANG 模块多版本通告

配置效果

在与 NETCONF 客户端能力交互时 (Hello 报文), 通告设备上所有支持的 yang 模块的所有版本。

注意事项

- 执行 **netconf yang multi-revision** 命令, 必须在 NETCONF 服务端能力报文 (Hello) 通告之前配置。
- 执行 **no netconf yang multi-revision** 命令, 必须在 NETCONF 服务端能力报文 (Hello) 通告之前配置, 且能力通告报文中一个 yang 模块只通告它当前最新版本。

配置方法

配置 YANG 模块多版本通告

- 可选配置。
- 必须在 NETCONF 服务端能力报文（Hello）通告之前，可选用此项配置。

【命令格式】 **netconf yang multi-revision**

【参数说明】 -

【缺省配置】 开启 YANG 模块多版本通告

【命令模式】 全局配置模式

【使用指导】 -

检验方法

-

配置举例

配置 YANG 模块多版本通告

【网络环境】

图 2-4

NETCONF服务端

NETCONF客户端



【配置方法】 ● 配置 YANG 模块多版本通告

NETCONF

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# netconf yang multi-revision
Ruijie(config)#
```

【检验方法】 -

常见错误

-

2.4.8 配置 NETCONF 优先支持 openconfig YANG 模型

配置效果

在 NETCONF 服务端启动时，设备决定加载 openconfig YANG 模型。

注意事项

- 执行 **netconf yang suite openconfig** 命令，必须设备重启才会生效，若配置支持 openconfig YANG 会支持 openconfig YANG 模型，否则支持锐捷自定义 YANG 模型。
- 执行 **no netconf yang suite openconfig** 命令，必须设备重启才会生效，配置生效后不会支持 openconfig YANG 模型。
- 该命令设备重启生效，show running-config 看到的是当前执行 CLI 后的配置。

配置方法

配置 NETCONF 优先支持 openconfig YANG 模型

- 可选配置。
- 必须设备重启才会生效。

【命令格式】 **netconf yang suite openconfig**

【参数说明】 -

【缺省配置】 关闭支持 openconfig YANG 模型

【命令模式】 全局配置模式

【使用指导】 -

检验方法

- 使用 **show netconf yang-suite** 命令，可以查看 NETCONF 当前生效的 YANG 模型配置及当前配置的 YANG 模型（可能未生效，需要重启设备）。

配置举例

配置 NETCONF 优先支持 openconfig YANG 模型

【网络环境】

图 2-4

NETCONF服务端

NETCONF客户端



【配置方法】

- 配置 YANG 模块多版本通告

NETCONF

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
```



```
Ruijie(config)#netconf yang suite openconfig
This command will take effect after device restart.
Ruijie(config)#
```

【检验方法】 -

常见错误

-

2.5 监视与维护

查看运行情况

作用	命令
显示 NETCONF 当前 yang 模型	show netconf yang-suite

3 RMON

3.1 概述

RMON 全称是 Remote Network Monitoring，远端网络监控。

RMON 用来解决从一个中心点管理各局域分网和远程站点的问题。RMON 中，网络监视数据包含了一组统计数据和性能指标，这些数据可以用来监控网络利用率，以用于网络规划，性能优化和协助网络错误诊断。

RMON 适主要用于管理设备向被监控管理设备进行远程监控管理。

协议规范

STD 0059 / RFC 2819 : Remote Network Monitoring Management Information Base

RFC4502 : Remote Network Monitoring Management Information Base Version 2

RFC 3919 : Remote Network Monitoring (RMON) Protocol Identifiers for IPv6 and Multi Protocol Label Switching (MPLS)

RFC 3737 : IANA Guidelines for the Registry of Remote Monitoring (RMON) MIB Modules

RFC 3434 : Remote Monitoring MIB Extensions for High Capacity Alarms

RFC 3395 : Remote Network Monitoring MIB Protocol Identifier Reference Extensions

RFC 3287 : Remote Monitoring MIB Extensions for Differentiated Services

RFC 3273 : Remote Network Monitoring Management Information Base for High Capacity Networks

RFC 2896 : Remote Network Monitoring MIB Protocol Identifier Macros

RFC 2895 : Remote Network Monitoring MIB Protocol Identifier Reference

3.2 典型应用

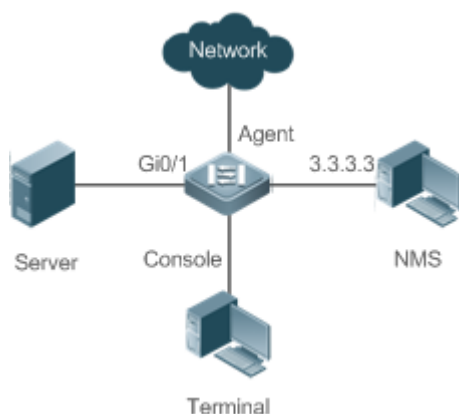
典型应用	场景描述
统计监控接口信息	应用 RMON 的四个功能于接口监管的网络通信

3.2.1 统计监控接口信息

应用场景

用户通过 RMON 以太网统计功能监管接口的累计信息，通过历史统计功能监管接口每个监管间隔时间内的接口报文数信息，使用告警功能即时获知接口报文数异常情况。组网图如下所示：

图 3-1



功能部署

对接口进行监管，分别累加统计接口报文数信息，统计接口监管时间间隔内的报文数信息以及带宽利用率，如果接口报文数异常，告警通知网管，配置要点如下：

- 在接口下配置 RMON 以太网统计功能；
- 在接口下配置 RMON 历史统计功能；
- 在配置模式下配置 RMON 告警表以及定义相应的 RMON 事件处理动作，告警监管的对象为接口下配置的 RMON 以太网统计表的具体字段 OID 值。

3.3 功能详解

基本概念

RMON 定义了多个 RMON 组，锐捷产品支持其中的统计组、历史组、告警组、事件组。下面对四个组做简要的介绍：

统计组

统计组用于对以太网接口的流量信息进行监控、统计，是从创建表项起到当前阶段的累加值，统计的内容包括丢弃的数据包、广播数据包、CRC 错误、大小块、冲突等，统计结果将保存在以太网统计表中以便管理员随时查看。

历史组

历史组(History)用于定期收集网络流量信息，记录每一个周期内的网络流量信息的累加值以及带宽利用率，并保存在历史控制表中以便管理员日后处理，它包含两个小组：

- HistoryControl 组用来设置采样间隔时间、采样数据源等控制信息。
- EthernetHistory 组为管理员提供有关网段流量、错误包、广播包、利用率以及碰撞次数等统计信息的历史数据。

告警组

警报组(Alarm)用于监控指定的 MIB(Management Information Base，管理信息库)对象，当这个 MIB 对象的值超过设定的上限值或低于设定的下限值时，会触发警报，警报被当作事件来处理。

事件组

事件组（Event）用于定义事件的处理方式。当监控的 MIB 对象达到告警条件时，就会触发事件，事件有如下四种处理方式：

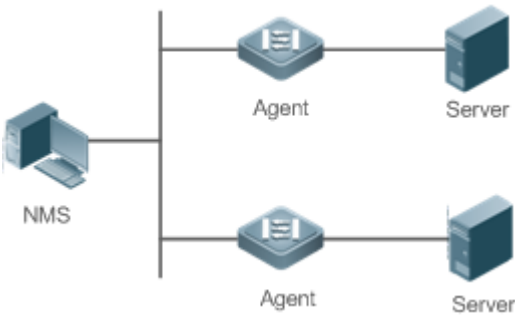
- none：不做任何动作。
- log：将事件相关信息记录在日志记录表中，以便管理员随时查看。
- snmp-trap：向网管站发送 Trap 消息告知该事件的发生。
- log-and-trap：将事件相关信息记录在日志记录表中，同时向网管站发送 Trap 消息。

工作原理

RMON 允许有多个监控者，可以通过两种方法进行数据收集：一种方法是利用专用的 RMON probe（RMON 探测仪）收集数据，NMS（网络管理系统）可以直接从 RMON probe 获取 RMON MIB 的全部信息。另一种方法是将 RMON Agent 植入网络设备（如交换机，路由器等），使设备具备 RMON probe 功能。NMS 用 SNMP 的基本命令与其交换数据信息，收集网络管理信息，但这种方式受设备资源的限制，一般不能获取 RMON MIB 的所有数据，一般只收集四个组信息。

下图给出了 NMS 与 RMON 代理通信的例子。通过运行在设备上的 RMON Agent，NMS 可以获取与被管网络设备接口的网段上的整体流量、错误统计和性能统计等信息，从而实现对网络设备的远程管理。

图 3-2



功能特性

功能特性	作用
RMON 以太网统计功能	对监控的以太网接口报文数、字节数等数据进行累加统计。
RMON 历史统计功能	记录以太网接口在配置的间隔时间内通信的报文数、字节数等数据进行统计，并计算间隔时间内的带宽利用率。。
RMON 告警功能	告警表与事件表结合使用，间隔对监控的变量的值进行采样，触及上下限就触发相关事件表做事件处理，或者不做任何处理。

3.3.1 RMON 以太网统计功能

工作原理

累加统计从创建表项起到现阶段的以太网接口的网络流量信息。

相关配置

配置 RMON 统计项

- 缺省情况下，RMON 以太网统计功能关闭。
- 使用 **rmon collection stats** 命令在指定的以太网接口上创建以太网统计表项。
- 在指定接口下创建统计表项成功后，统计组就对当前接口的流量信息进行统计，它统计的是 RMON 以太网统计表定义的变量，记录的是 RMON 统计表创建起至当前阶段时间内变量的累加值。

3.3.2 RMON 历史统计功能

工作原理

记录每一个周期内的以太网接口流量信息的累加统计值。

相关配置

配置 RMON 历史控制表项

- 缺省情况下，配置 RMON 历史统计功能关闭。
- 使用 **rmon collection history** 命令在以太网接口上创建历史控制表项。
- RMON 历史组统计的是 RMON 历史表定义的变量，记录的是每个周期内变量的累加值。

3.3.3 RMON 告警功能

工作原理

周期性地监报告警变量值的变化，如果告警变量值触及指定的上下限阈值，则触发相应的事件处理，如发送 trap 信息，或者产生一条 logTable 表项记录等。但连续多次触有上限阈值或者下限阈值，只触发相应的事件处理一次，等待触发相反阈值处理。

相关配置

配置事件表

- 缺省情况下，配置 RMON 事件组功能关闭。
- 使用 **rmon event** 命令配置事件表。

配置告警表项

- 缺省情况下，配置 RMON 告警组功能关闭。
- 使用 **rmon event** 命令配置事件表、**rmon alarm** 命令配置 RMON 告警表。
- RMON 告警功能是由告警表和事件表共同实现。如果告警事件需要向管理设备发送 Trap 信息的话，则必须事先保证 SNMP Agent 已经正确配置。SNMP Agent 的配置请参见《SNMP 配置指南》。
- 如果配置的告警对象是 RMON 统计组或者历史组的某一段节点，需要先在被监控的以太网接口下配置 RMON 统计功能或者 RMON 历史统计功能。

3.4 配置详解

配置项	配置建议 & 相关命令	
配置 RMON 以太网统计功能	 必须配置。用于累加统计以太网接口流量信息。	
	rmon collection stats	配置以太网统计表项。
配置 RMON 历史统计功能	 必须配置。用于间隔统计间隔时间内的以太网接口流量信息以及带宽使用率。	
	rmon collection history	配置历史控制表项。
配置 RMON 告警功能	 必须配置。用于监测某一变量的数据变化是否在合法范围内。	
	rmon event	配置事件表项。
	rmon alarm	配置告警表项。

3.4.1 配置 RMON 以太网统计功能

配置效果

可以获知被监控的以太网接口从创建表项起到现阶段的流量信息的累加统计值。

注意事项

不允许批量接口配置，即不允许在批量接口配置模式下进行配置。

配置方法

配置 RMON 统计项

- 必选配置。
- 如果需要对指定接口进行统计、监控，必须在该接口下配置以太网统计表项。

检验方法

使用 **show rmon stats** 命令可以查看以太网统计信息。

相关命令

配置 RMON 统计项

【命令格式】 **rmon collection stats index [owner ownername]**

【参数说明】 *index*：统计表项的索引号，取值范围：1~65535；

owner ownername：设置表项的创建者 *ownername*，*ownername* 为 1~63 个字符的字符串，区分大小写。

【命令模式】 接口模式

【使用指导】 不允许对已经配置的统计表项参数进行修改。

配置举例

配置 RMON 以太网统计功能

【网络环境】

图 3-3



如上图所示，RMON Agent 与 Server 服务器连接，网管需要通过 RMON 统计组来对 G0/1 接口的接收报文进行性能统计，以便随时通过查看数据了解相应接口接收报文的数据，及时对异常网络情况采取措施处理。

【配置方法】 ● 在接口 GigabitEthernet 0/3 上配置统计表实例，对该接口进行流量统计。

Agent

```
Ruijie# configure terminal
Ruijie (config)# interface gigabitEthernet 0/3
Ruijie (config-if-GigabitEthernet 0/3)# rmon collection stats 1 owner admin
```

【检验方法】 通过 **show rmon stats** 查看以太网统计信息。

Agent

```
Ruijie# show rmon stats
ether statistic table:
```

```
index = 1
interface = GigabitEthernet 0/1
owner = admin
status = 1
dropEvents = 0
octets = 25696
pkts = 293
broadcastPkts = 3
multiPkts = 0
crcAlignErrors = 0
underSizePkts = 0
overSizePkts = 0
fragments = 0
jabbers = 0
collisions = 0
packets64Octets = 3815
packets65To127Octets = 1695
packets128To255Octets = 365
packets256To511Octets = 2542
packets512To1023Octets = 152
packets1024To1518Octets = 685
```

常见错误

重复配置或者修改已经配置的统计表表项。

3.4.2 配置 RMON 历史统计功能

配置效果

可以获知被监控的以太网接口的每一个周期内的流量信息累加统计值及带宽利用率。

注意事项

不允许批量接口配置，即不允许在批量接口配置模式下进行配置。

配置方法

- 必选配置。
- 如果需要对指定接口收集网络统计信息，必须在接口上配置 RMON 历史控制表项。

检验方法

使用 **show rmon history** 命令可以查看历史组统计信息。

相关命令

配置 RMON 历史控制表项

- 【命令格式】 **rmon collection history index [owner ownername] [buckets bucket-number] [interval seconds]**
- 【参数说明】 **index**：历史统计表项的索引号，取值范围：1~65535
- owner ownername**：设置表项的创建者 *ownername*，*ownername* 为 1~63 个字符的字符串，区分大小写。
- buckets bucket-number**：设置历史统计表项对应的历史表容量，即设置历史表最多可容纳的记录数 *bucket-number*，*bucket-number* 取值范围：1~65535，默认值是 10
- interval seconds**：设置统计周期值 *seconds*，单位为秒，*seconds* 取值范围：1~3600，默认值是 1800s
- 【命令模式】 接口模式
- 【使用指导】 不允许对已经配置的历史统计表项参数进行修改。

配置举例

配置 RMON 历史统计功能

【网络环境】

图 3-4



如上图所示，RMON Agent 与 Server 服务器连接，网管需要通过 RMON 历史组来对 G0/1 接口的接收报文进行周期性统计，周期时间为 60 秒，从而达到对网络的监控，掌握突发情况数据。

- 【配置方法】
- 在接口 GigabitEthernet 0/3 上配置历史控制表，对该接口进行周期性流量统计
- Agent
- ```
Ruijie# configure terminal
Ruijie(config)# interface gigabitEthernet 0/3
Ruijie(config-if-GigabitEthernet 0/3)# rmon collection history 1 buckets 5 interval 300 owner admin
```
- 【检验方法】 通过 **show rmon history** 查看历史组统计信息。
- Agent
- ```
Ruijie# show rmon history
rmon history control table:
      index = 1
```

```
interface = GigabitEthernet 0/1
bucketsRequested = 5
bucketsGranted = 5
interval = 60
owner = admin
stats = 1
```

rmon history table:

```
index = 1
sampleIndex = 786
intervalStart = 6d:18h:37m:38s
dropEvents = 0
octets = 2040
pkts = 13
broadcastPkts = 0
multiPkts = 0
crcAlignErrors = 0
underSizePkts = 0
overSizePkts = 0
fragments = 0
jabbers = 0
collisions = 0
utilization = 0
```

```
index = 1
sampleIndex = 787
intervalStart = 6d:18h:38m:38s
dropEvents = 0
octets = 1791
pkts = 16
broadcastPkts = 1
multiPkts = 0
crcAlignErrors = 0
underSizePkts = 0
overSizePkts = 0
fragments = 0
jabbers = 0
collisions = 0
utilization = 0
```

```
index = 1
```

```
sampleIndex = 788
intervalStart = 6d:18h:39m:38s
dropEvents = 0
octets = 432
pkts = 6
broadcastPkts = 0
multiPkts = 0
crcAlignErrors = 0
underSizePkts = 0
overSizePkts = 0
fragments = 0
jabbers = 0
collisions = 0
utilization = 0

index = 1
sampleIndex = 789
intervalStart = 6d:18h:40m:38s
dropEvents = 0
octets = 432
pkts = 6
broadcastPkts = 0
multiPkts = 0
crcAlignErrors = 0
underSizePkts = 0
overSizePkts = 0
fragments = 0
jabbers = 0
collisions = 0
utilization = 0

index = 1
sampleIndex = 790
intervalStart = 6d:18h:41m:38s
dropEvents = 0
octets = 86734
pkts = 934
broadcastPkts = 32
multiPkts = 23
crcAlignErrors = 0
underSizePkts = 0
```

```
overSizePkts = 0
fragments = 0
jabbers = 0
collisions = 0
utilization = 0
```

常见错误

重复配置或者修改已经配置的历史控制表表项。

3.4.3 配置 RMON 告警功能

配置效果

周期性监控告警变量的值变化是否在指定的合法范围内。

注意事项

如果触发告警事件时，需要向管理设备发送 Trap 信息的话，必须保证 SNMP Agent 已经正确配置。SNMP Agent 配置请参见《SNMP 配置指南》。

如果告警变量是 RMON 统计组或者是历史组中定义的 MIB 变量时，必须在被监控的以太网接口上配置 RMON 以太网统计功能或者 RMON 历史统计功能，否则创建告警表失败。

配置方法

📄 配置事件表项

- 必须配置。
- 在全局配置模式下配置

📄 配置告警表项

- 必须配置。
- 在全局配置模式下配置

检验方法

- 使用 **show rmon event** 查看事件表信息。
- 使用 **show rmon alarm** 查看告警表信息。

相关命令

配置事件表

- 【命令格式】 **rmon event** *number* [**log**] [**trap** *community*] [**description** *description-string*] [**owner** *ownername*]
- 【参数说明】 *number*：事件表的索引号，取值范围：1~65535。
log：日志事件，当事件被触发时，系统会记录日志。
trap *community*：Trap 事件，当事件被触发时，系统会以 *community* 为团体名发送 Trap。
description *description-string*：设置事件的描述信息 *description-string*，*description-string* 为 1~127 个字符的字符串。
owner *ownername*：设置表项创建者 *ownername*，*ownername* 为 1~63 个字符的字符串，区分大小写。
- 【命令模式】 全局配置模式
- 【使用指导】 允许对已经配置的事件表项参数进行修改，包括事件类型、Trap 团体名、事件描述、事件创建者等。

配置 RMON 告警组

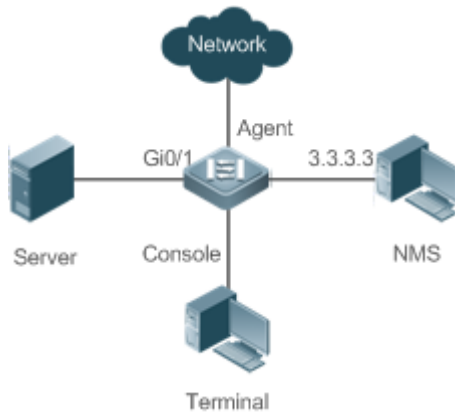
- 【命令格式】 **rmon alarm** *number* *variable* *interval* {**absolute** | **delta**} **rising-threshold** *value* [*event-number*] **falling-threshold** *value* [*event-number*] [**owner** *ownername*]
- 【参数说明】 *number*：告警表项的索引号，限值范围：1~65535。
variable：告警变量，为 1~255 个字符的字符串，并且以节点 OID 的点分格式（格式为 entry.integer.instance，如 1.3.6.1.2.1.2.1.10.1）进行表示。
Interval：采样间隔时间，单位为秒，取值范围为 1~2147483647。
absolute：采样类型为绝对值采样，即采样时间到达时直接提取变量的值。
delta：采样类型为变化值采样，即采样时间到达时提取的是变量在采样间隔内的变化值。
rising-threshold *value*：设置采样数量的上限参数 *value*，取值范围：-2147483648~+2147483647。
event-number：到达上下限时触发事件号为 *event-number* 的事件。
falling-threshold *value*：设置采样数量的下限参数 *value*，取值范围：-2147483648~+2147483647。
owner *ownername*：设置表项的创建者 *ownername*，*ownername* 为 1~63 个字符的字符串，区分大小写。
- 【命令模式】 全局配置模式
- 【使用指导】 允许对已经配置的告警表项参数进行修改，包括告警变量、采样类型、表项的创建者、采样间隔时间、采样数量的上/下限值及其对应的触发事件。

配置举例

配置 RMON 告警功能

【网络环境】

图 3-5



假设 NMS 上运行 SNMPV1，访问设置时使用的团体名为 public，属性为可读写，NMS 接收 trap 的 IP 地址为 3.3.3.3。

假设监控接口 GigabitEthernet0/3 上接收到的未知协议的报文数，对应的 OID 值是 1.3.6.1.2.1.2.2.1.15.3，采样方式为相对采样，采样间隔时间为 60 秒，当相对采样值超过 100 时或者低于 10 时，分别触发事件 1 和事件 2，事件 1 发 trap 信息和 log 信息，事件 2 只生成日志记录表。

RMON Agent 通过终端 terminal 完成相关配置，与 NMS 设备连接通信，Gi0/1 跟服务器 Server 连接，现需要监控 Gi0/1 上收到未知协议的报文数。采样间隔时间 60 秒，绝对采样值小于 10 时，只记录 log，而大于 100 时，则需要记录 log 和发送 trap 给 NMS。

【配置方法】

- 配置 SNMP 主机接收告警功能发送 trap。
- 配置事件组动作来处理告警触发情况。
- 配置告警功能。

Agent

```

Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# snmp-server community public rw
Ruijie(config)# snmp-server host 3.3.3.3 trap public
Ruijie(config)# rmon event 1 description rising-threshold-event log trap public owner admin
Ruijie(config)# rmon event 2 description falling-threshold-event log owner admin
Ruijie(config)# rmon alarm 1 1.3.6.1.2.1.2.2.1.15.3 60 delta rising-threshold 100 1
falling-threshold 10 2 owner admin
  
```

【检验方法】

- 使用 **show rmon event** 查看事件表信息。
- 使用 **show rmon alarm** 查看告警表信息。

Agent

```

Ruijie# show rmon event
rmon event table:

    index = 1
    description = rising-threshold-event
    type = 4
    community = public
    lastTimeSent = 0d:0h:0m:0s
    owner = admin
  
```

```

        status = 1

        index = 2
        description = falling-threshold-event
        type = 2
        community =
        lastTimeSent = 6d:19h:21m:48s
        owner = admin
        status = 1

rmon log table:
        eventIndex = 2
        index = 1
        logTime = 6d:19h:21m:48s
        logDescription = falling-threshold-event

Ruijie# show rmon alarm
rmon alarm table:
        index: 1,
        interval: 60,
        oid = 1.3.6.1.2.1.2.2.1.15.3
        sampleType: 2,
        alarmValue: 0,
        startupAlarm: 3,
        risingThreshold: 100,
        fallingThreshold: 10,
        risingEventIndex: 1,
        fallingEventIndex: 2,
        owner: admin,
        stauts: 1
```

常见错误

- 输入监控的对象 OID 不合理，OID 对应的变量不存在或者类型不是整型或者无符号整型。
- 上限阈值小于等于下限阈值。

3.5 监视与维护

查看运行情况

作用	命令
----	----

查看所有 RMON 配置信息	show rmon
查看以太网统计表信息	show rmon stats
查看历史控制表信息	show rmon history
查看告警表信息	show rmon alarm
查看事件表信息	show rmon event

4 NTP

4.1 概述

NTP (Network Time Protocol , 网络时间协议) , 用来使网络设备时间同步化的一种应用层协议。它可以使网络设备对其服务器或时钟源做同步化, 提供高精度度的时间校正 (LAN 上与标准时间差小于 1 毫秒 , WAN 上几十毫秒) , 且可使用加密确认的方式来防止攻击。

目前锐捷设备支持 NTP 的客户端与服务器功能, 即设备既可以从时间服务器上同步时间, 也能够作为时间服务器对其他设备进行时间同步。在作为服务器工作时设备仅支持单播 Server 模式。

协议规范

- RFC 1305 : Network Time Protocol (Version 3)

4.2 典型应用

典型应用	场景描述
基于外部时钟参考源同步时间	设备即作为客户端从外部时钟源同步时间, 同步成功后又作为服务器向其他设备提供时间同步服务。
基于本地时钟参考源同步时间	设备将本地时钟作为 NTP 可靠参考时钟源, 作为服务器向其它设备提供时间同步服务。

4.2.1 基于外部时钟参考源同步时间

应用场景

如图所示：

- DEVICE-A 作为可靠参考时钟源对外提供时间同步服务
- DEVICE-B 指定 DEVICE-A 为 NTP 服务器，从 DEVICE-A 同步时间。
- DEVICE-B 同步成功后向 DEVICE-C 提供时间同步服务。

图 3-1



功能部属

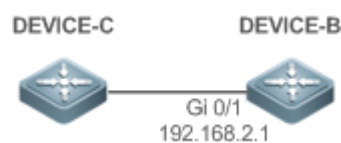
将 DEVICE-B 配置为 NTP 外部时钟参考模式

4.2.2 基于本地时钟参考源同步时间

应用场景

如图所示，DEVICE-B 将本地时钟作为 NTP 参考时钟源，向 DEVICE-C 提供时间同步服务。

图 3-2



功能部属

将 DEVICE-B 配置为 NTP 本地时钟参考模式。

4.3 功能详解

基本概念

📄 NTP 报文

根据 RFC1305 定义，NTP 采用 UDP 报文进行传输，UDP 端口号为 123。

NTP 时间同步报文格式如图 1-3

图 3-3 NTP 时间同步报文格式

0	7	15	23	31
LI	VN	Mode	Stratum	Poll Interval
Precision				
Root Delay (32-bit)				
Root Dispersion (32-bit)				
Reference Clock Identifier (32-bit)				
Reference Timestamp (64-bit)				
Originate Timestamp (64-bit)				
Receive Timestamp (64-bit)				
Transmit Timestamp (64-bit)				
Authenticator (optional 96-bit)				

- Leap Indicator (LI): 2 比特, 闰秒标志。

i 00-无警告信息 01-上一分钟有 61 秒 10-上一分钟有 59 秒 11-时钟未同步

- Version Number (VN): 3 比特, NTP 版本号, 当前版本号为 3。
- Mode : 3 比特, NTP 工作模式。

i 0-未定义 1-主动对等体 2-被动对等体 3-客户端 4-服务器 5-广播 6-控制信息 7-保留

- Stratum : 8 比特, 本地时钟的层数 (0-未定义 1-主参考时钟源 其它值-次参考时钟源)。
- Poll Interval : 8 位整数, 轮询时间 (秒数)
- Precision : 8 位整数, 本地时钟的时间精度 (秒数)
- Root Delay : 32 位整数, 到主参考时钟源的往返时间
- Root Dispersion : 32 位整数, 相对于主参考时钟源的最大误差
- Reference Clock Identifier : 32 比特, 参考时钟源的标识
- Reference Timestamp : 64 位时间戳, 最后一次被设置或者被校正的时间
- Originate Timestamp : 64 位时间戳, 时间同步请求报文离开客户端的本地时间
- Receive Timestamp : 64 位时间戳, 时间同步请求报文到达服务器的本地时间
- Transmit Timestamp : 64 位时间戳, 时间同步响应报文离开服务器的本地时间
- Authenticator (可选): 验证信息

📌 NTP 服务器

设备将本地时钟作为参考时钟源, 为网络中的其它设备提供时间同步服务。

📌 NTP 客户端

设备作为 NTP 客户端从网络中的 NTP 服务器同步时间。

层数 (stratum)

NTP 使用 “层数 (stratum)” 的概念来描述设备距离权威时钟源的 “跳数 (hops)”。一个层数为 1 的时间服务器应当有个直连的原子钟或电波钟；层数为 2 的时间服务器就从层数为 1 的服务器获取时间；层数为 3 的服务器就从层数为 2 的获取时间..... 如此递推。因此时钟层数数值更低的时钟源即被认为拥有更高的时钟精度。

硬件时钟

硬件时钟根据设备上的石英晶体振荡器频率工作，由设备的电池为其供电，设备关机后硬件时钟依然运行。在设备启动运行后，会从硬件时钟读取时间信息，作为设备的软件时间。

功能特性

功能特性	作用
NTP 时间同步	使网络设备根据其服务器或可靠时钟源进行时间同步，以实现高精度度的时间校正。
NTP 安全认证	通过 NTP 报文加密认证方式，防止非可靠时钟源对设备进行时间同步干扰。
NTP 访问控制	根据访问控制列表对收到的 NTP 报文进行源过滤。

4.3.1 NTP 时间同步

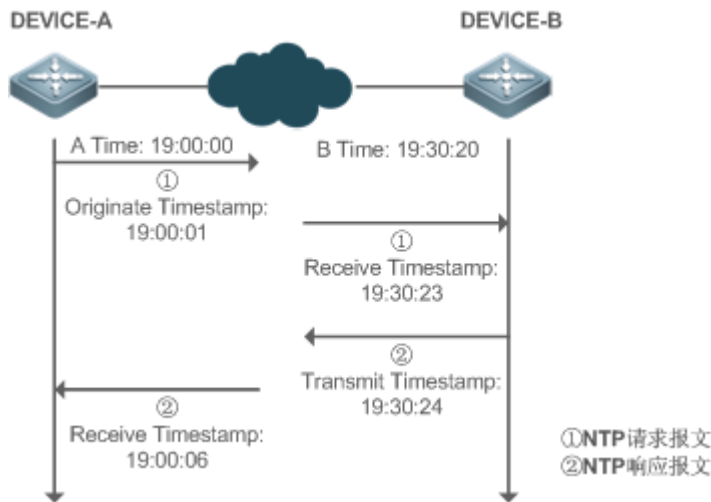
工作原理

NTP 同步时间的方式是通过客户端与服务器之间交互 NTP 报文：

- 客户端每隔 64 秒钟向所有服务器发送时间同步报文。收到服务器响应报文后，对所有服务器的响应报文进行过滤和选择，最后和优选服务器的时间进行同步。
- 服务器收时间同步请求报文时，将本地时钟作为参考源，按协议要求将本地时间信息填充到响应报文返回给客户端。

NTP 时间同步报文格式如图 1-4

图 3-4 NTP 基本工作原理图



DEVICE-B (下面简称 B) 作为 NTP 参考时钟源, DEVICE-A (下面简称 A) 作为 NTP 客户端从 DEVICE-B 同步时间, 在某一时刻 A 的本地时钟为 19:00:00, B 的本地时钟为 19:30:20 :

4. A 发出 NTP 请求报文, 报文离开 A 的本地时间 (T0) 为 19:00:00, 填充在 Originate Timestamp
5. 经过 2 秒的网络延时, B 收到请求报文的本地时间 (T1) 为 19:30:23, 填充在 Receive Timestamp
6. B 处理 NTP 请求, 1 秒后响应 NTP 报文, 报文离开 B 的本地时间 (T2) 为 19:30:24, 填充在 Transmit Timestamp
7. 经过 2 秒的网络延时, A 接收到响应报文, 响应报文到达 A 的本地时间 (T3) 为 19:00:06

时间同步的具体算法如下 :

- A 通过公式 $((T1-T0)+(T2-T3))/2$ 计算出 B 和 A 的时间差为 30 分 20 秒
- A 通过公式 $(T3-T0)-(T2-T1)$ 计算出 A 和 B 的报文往返的延时为 4 秒

📌 NTP 工作模式

- 外部时钟参考模式

在该模式下, 设备即充当服务器又充当客户端, 如果收到来自其它客户端发出的时间同步请求, 必须先从指定服务器同步时间, 同步成功后才可以向其它客户端提供时间同步服务。

- 本地时钟参考模式

在该模式下, 设备默认本地时钟即为可靠时钟源, 直接向其它客户提供时间同步服务。

相关配置

📌 配置 NTP 服务器

- 缺省情况下, NTP 功能关闭。
- 通过 **ntp server** 命令指定 NTP 服务器 (即外部时钟参考源), 即可开启 NTP 功能。
- 配置后设备处于外部时钟参考模式。

📌 实时同步

- 缺省情况下, 设备每隔 64 秒进行一次时间同步。

📌 更新硬件时钟

- 缺省情况下, 设备同步完时间后不会把时间更新到硬件时钟。
- 配置 **ntp update-calendar** 命令可以使设备每次时间同步成功时会自动更新硬件时钟。

📌 配置 NTP 主时钟

- 缺省情况下, 设备处于外部时钟参考模式。
- 通过 **ntp master** 命令可以将设备配置为本地时钟参考模式。

4.3.2 NTP 安全认证

为防止对时间服务器的恶意破坏，NTP 使用了识别(Authentication)机制，检查时间同步信息是否是真正来自所宣称的服务器并检查资料的返回路径，以提供对抗干扰的保护机制。

工作原理

NTP 客户端和服务端配置相同的密钥。发送请求报文和响应报文时，设备根据指定的密钥和 NTP 报文内容采用 MD5 算法计算出报文的哈希值填充到报文的认证信息。接收设备根据认证信息判断是否报文发送端是否可信的设备或者报文是否被篡改。

相关配置

配置 NTP 全局安全认证机制

- 缺省情况下，没有开启 NTP 安全认证机制。
- 通过 `ntp authenticate` 命令可开启 NTP 安全认证机制。

配置 NTP 全局认证密钥

- 缺省情况下，没有配置全局认证密钥。
- 通过 `ntp authentication-key` 命令可开启 NTP 安全认证机制。

配置 NTP 全局信任密钥 ID

- 缺省情况下，没有配置全局信任密钥。
- 通过 `ntp trusted-key` 命令设备作为参考时钟源对外提供时间同步服务的信任密钥。

配置外部参考时钟源的信任密钥 ID

- 通过 `ntp server` 指定外部参考时钟源的同时可以指定该时钟源的信任密钥。

4.3.3 NTP 访问控制

工作原理

通过 ACL 提供了一种最小限度的安全措施

相关配置

配置 NTP 服务的访问控制权限

- 缺省情况下，没有 NTP 访问控制权限。

- 通过 `ntp access-group` 可配置 NTP 的访问控制权限。

4.4 产品说明



锐捷目前的版本只支持最大 1024 认证密钥，每个服务器允许设置唯一——一个密钥进行安全通信。

4.5 配置详解

配置项	配置建议&相关命令	
配置 NTP 基本功能	⚠ 必须配置，用于开启 NTP 功能，开启后设备处于外部时钟参考模式。	
	<code>ntp server</code>	配置 NTP 服务器
	<code>ntp update-calendar</code>	自动更新硬件时钟
	⚠ 可选配置，用于将设备配置为本地时钟参考模式。	
	<code>ntp master</code>	配置 NTP 主时钟
	⚠ 可选配置，用于关闭 NTP 功能。	
	<code>no ntp</code>	关闭所有 NTP 功能，清空 NTP 配置。
	<code>ntp disable</code>	禁止接收指定接口的 NTP 报文
	<code>ntp service disable</code>	关闭 NTP 对外提供时间同步服务功能
配置 NTP 安全认证	⚠ 可选配置，用于防止非可靠时钟源对设备进行时间同步干扰。	
	<code>ntp authenticate</code>	开启安全认证机制
	<code>ntp authentication-key</code>	设置安全认证全局密钥
	<code>ntp trusted-key</code>	配置时间同步服务可信密钥
	<code>ntp server</code>	配置外部参考时钟源的可信密钥
配置 NTP 访问控制	⚠ 可选配置，用于对收到的 NTP 报文进行源过滤。	
	<code>ntp access-group</code>	设置 NTP 的访问控制权限

4.5.1 配置 NTP 基本功能

配置效果

外部时钟参考模式

- 设备作为客户端，从外部参考时钟源同步时间到本地时钟
- 时间同步成功后，设备可作为时间同步服务器，对外提供时间同步服务

本地时钟参考模式

- 设备的本地时钟作为 NTP 参考时钟源，对外提供时间同步服务

注意事项

- 客户端/服务器模式，设备只有从外部的可靠时钟源同步成功后，才能作为时间同步服务器对外提供服务。
- 一旦配置本地时钟参考模式，系统便不会与比其时钟层数数值更高的时钟源进行同步。
- 将本地时钟设置为主时钟（尤其是指定了较低的时钟层数值时）很有可能将真正有效时钟源覆盖。如果对同一网络中的多个设备都使用了该命令，则可能由于设备之间的时钟差异导致网络的时钟同步不稳定。
- 将本地时钟设置为主时钟前，如果系统从未与外部时钟源同步过，则有可能需要手动校准系统时钟以保证其不会有过大的偏差（关于如何手动校准系统时钟请参考配置指南中的系统时间配置部分）。

配置方法

配置 NTP 服务器

- 必须配置，至少指定一个外部参考时钟源（最多可配置 20 个不同的外部参考时钟源）。
- 如果需要关联配置 NTP 密钥，在配置 NTP 服务器前，必须先配置 NTP 安全认证。

自动更新硬件时钟

- 可选配置
- 默认情况下，时间同步成功后只更新系统时钟，不会更新硬件时钟。
- 配置此命令，时间同步成功后会自动更新硬件时钟。

配置 NTP 主时钟

- 如果需要将设备切换到本地时钟参考模式，可通过此命令。

关闭 NTP 功能

- 如果需要关闭 NTP 功能，并且清空 NTP 配置，可通过 **no ntp** 命令
- 默认情况，开启 NTP 功能后所有接口都可以接收 NTP 报文。如果需要禁止特定接口的 NTP 功能时可通过 **ntp disable** 命令。

关闭 NTP 对外提供时间同步服务功能

- NTP 处于客户端/服务器模式，设备从外部的可靠时钟源同步时间成功后，会作为时间同步服务器对外提供时间同步服务。如果只想让 NTP 仅作为客户端使用，则需要配置 **ntp service disable**，关闭 NTP 对外提供时间同步服务功能。

检验方法

- 通过 **show ntp status** 查看 NTP 配置信息。

- 通过 **show clock** 查看是否完成时间同步

相关命令

配置 NTP 服务器 **ntp server**[vrf vrf-name][ip-addr | domain | ip domain | ipv6 domain][version version][source if-name][key keyid][prefer]

【参数说明】 oob：参考时钟源是否绑定 MGMT 口

vrf-name：参考时钟源绑定 VRF 的名称

ip-addr：参考时钟源的 IPv4/IPv6 地址

domain：参考时钟源的 IPv4/IPv6 域名

version：NTP 版本号，取值为 1-3。

if-name：接口类型，包括 AggregatePort、Dialer、GigabitEthernet、Loopback、Multilink、Null、Tunnel、Virtual-ppp、Virtual-template、Vlan 类型。

keyid：同参考时钟源通信采用的密钥(1-4294967295)

prefer：参考时钟源是否高优先级

mgmt-name：指定在 oob 模式下报文的出口管理口。

【命令模式】 全局模式

【使用指导】 在缺省情况下，没有配置 NTP 服务器。锐捷的客户端系统支持最多同时与 20 个 NTP 服务器交互，（在全局认证以及密钥相关设置完成后）可以为每一个服务器设置一个认证密钥，发起与服务器的加密通信。

 如果需要设置认证密钥，在配置 NTP 服务器前必须先配置 NTP 安全认证。

与服务器的默认通信版本为 NTP 版本 3，同时可以配置发送 NTP 报文的源接口，并只在发送接口上接收对应服务器的 NTP 报文。

更新硬件时钟

【命令格式】 **ntp update-calendar**

【参数说明】 -

【命令模式】 全局模式

【使用指导】 -

设置本地参考时钟源

【命令格式】 **ntp master**[stratum]

【参数说明】 stratum：指定本地时钟所处的层数，范围为 1～15；若不指定该参数则默认值为 8。

【命令模式】 全局模式

【使用指导】 -

关闭 NTP 功能

【命令格式】 **no ntp**

【参数说明】 -

【命令模式】 全局模式

【使用指导】 此命令可以快速关闭 NTP 所有功能，并且清空 NTP 所有配置

禁止接口接收 NTP 报文

- 【命令格式】 **ntp disable**
- 【参数说明】 -
- 【命令模式】 接口模式
- 【使用指导】 -

关闭 NTP 功能对外提供时间同步服务

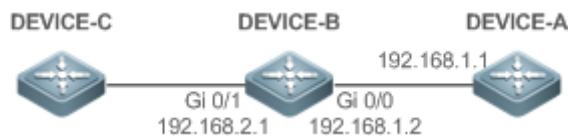
- 【命令格式】 **ntp service disable**
- 【参数说明】 -
- 【命令模式】 全局模式
- 【使用指导】 此命令可以关闭 NTP 对外提供时间同步服务功能，外部设备无法从设备同步到时间(此命令仅在部分版本上支持)。

配置举例

NTP 外部时钟参考模式

【网络环境】

图 1-5



- DEVICE-B：配置为 NTP 外部时钟参考模式
 - DEVICE-A：作为 DEVICE-B 的参考时钟源
 - DEVICE-C：从 DEVICE-B 同步时间
- 【配置方法】
- DEVICE-A 配置本地时钟为 NTP 参考时钟源
 - DEVICE-B 配置 DEVICE-A 为参考时钟源
 - DEVICE-C 配置 DEVICE-B 为参考时钟源

DEVICE-A

```
A#configure terminal
A(config)# ntp master
A(config)#exit
```

DEVICE-B

```
B#configure terminal
B(config)# ntp server 192.168.1.1
B(config)# exit
```

DEVICE-C

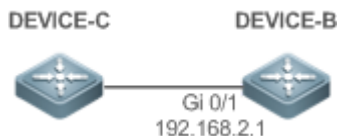
```
C#configure terminal
C(config)# ntp server 192.168.2.1
C(config)# exit
```

- 【检验方法】
- 在 DEVICE-B 上通过 **show ntp status** 查看 NTP 配置信息。
 - DEVICE-B 会向 192.168.1.1 发送时间同步报文，从 DEVICE-A 同步时间。
 - DEVICE-B 从 DEVICE-A 成功同步时间之后，可以响应 DEVICE-C 的时间同步请求。
 - 在 DEVICE-B 和 DEVICE-C 上通过 **show clock** 命令可以查看时间是否成功同步。

▾ NTP 本地时钟参考模式

【网络环境】

图 1-6



- DEVICE-B：本地时钟为 NTP 参考时钟源
 - DEVICE-C：从 DEVICE-B 同步时间
- 【配置方法】
- DEVICE-B 配置本地时钟为 NTP 参考时钟源
 - DEVICE-C 配置 DEVICE-B 为参考时钟源

DEVICE-B

```
B#configure terminal
B(config)# ntp master
B(config)# exit
```

DEVICE-C

```
C#configure terminal
C(config)# ntp server 192.168.2.1
C(config)# exit
```

【检验方法】

- 在 DEVICE-C 上通过 **show clock** 命令可以查看时间是否成功同步。

4.5.2 配置 NTP 安全认证

配置效果

▾ 从可信参考时钟源同步时间

设备作为客户端，只从可信任的外部参考时钟源同步时间到本地时钟

▾ 给可信设备提供时间同步服务

设备的本地时钟作为 NTP 参考时钟源，只对可信的设备提供时间同步服务

注意事项

客户端和服务器的认证密钥必须一致。

配置方法

▾ 配置 NTP 全局安全认证机制

- 必须配置
- 默认情况下设备不开启安全认证机制。

▾ 配置 NTP 全局认证密钥

- 必须配置
- 默认情况下设备没有认证密钥。

📌 配置 NTP 全局信任密钥 ID

- 可选配置
- 给可信设备提供时间同步服务，必须通过密钥 ID 指定可信认证密钥。
- 只允许配置一个信任密钥，所指定的认证密钥必须和可信设备一致。

📌 配置外部参考时钟源的认证密钥 ID

- 可选配置
- 从可信参考时钟源同步时间，必须通过密钥 ID 指定可信认证密钥。
- 每个可信参考时钟源分别对应一个认证密钥，认证密钥必须和可信参考时钟源的密钥一致。

检验方法

- 通过 **show run** 查看配置是否正确
- 通过 **show clock** 查看是否从可信设备同步时间

相关命令

📌 开启安全认证机制

【命令格式】 **ntp authenticate**

【参数说明】 -

【命令模式】 全局模式

【使用指导】 缺省情况下，客户端不使用全局安全识别机制。如果未使用安全识别机制则不对通信进行加密处理。但是仅仅设置了全局安全标志，并不代表一定采用了加密方式完成服务器与客户端的通信，还必须完成其他全局密钥配置并设置服务器加密密钥才可能发起和服务器的加密通信。

📌 设置全局认证密钥

【命令格式】 **ntp authentication-key key-id md5 key-string [enc-type]**

【参数说明】 *key-id* : 认证密钥的全局 ID (1-4294967295)。

key-string : 密钥字符串。

enc-type : 可选。输入的密钥是否加密 (0 表示无加密，7 表示简单加密，默认为无加密)。

【命令模式】 全局模式

【使用指导】 -

📌 设置 NTP 服务的可信密钥

- 【命令格式】 `ntp trusted-key key-id`
- 【参数说明】 `key-id` : 认证密钥的全局 ID (1-4294967295)。
- 【命令模式】 全局模式
- 【使用指导】 -

设置外部参考时钟源的可信密钥

参考 [“配置 NTP 服务器”](#)

配置举例

安全认证

【网络环境】

图 1-7



- DEVICE-B : 配置为 NTP 客户端/服务器模式，给 DEVICE-C 提供需要安全认证的 NTP 服务，认证密钥为 “abcd”
- DEVICE-A : 作为 DEVICE-B 的参考时钟源
- DEVICE-C : 从 DEVICE-B 同步时间
- DEVICE-B 配置 DEVICE-A 为参考时钟源
- DEVICE-C 配置 DEVICE-B 为参考时钟源

DEVICE-B

```
B#configure terminal
B(config)# ntp authentication-key 1 md5 abcd
B(config)# ntp trusted-key 1
B(config)# ntp server 192.168.1.1
B(config)# exit
```

DEVICE-C

```
C#configure terminal
C(config)# ntp authentication-key 1 md5 abcd
C(config)# ntp server 192.168.2.1 key 1
C(config)# exit
```

- 【检验方法】
- DEVICE-B 会向 192.168.1.1 发送时间同步报文，携带认证信息，从 DEVICE-A 同步时间。
 - 在 DEVICE-B 上通过 `show clock` 命令查看时间是否成功同步。

配置举例

4.5.3 配置 NTP 访问控制

配置效果

NTP 服务的访问控制功能提供了一种最小限度的安全措施（更安全的方法是使用 NTP 身份验证机制）。

注意事项

- 目前系统暂未支持控制查询功能，用于通过网络管理设备对 NTP 服务器进行控制（如设置闰秒标记或监控其工作状态等）。虽然是按照上述顺序进行规则匹配，但涉及到与控制查询相关的请求都无法支持。
- 如果未配置任何访问控制规则，则所有访问都是允许的。但一旦配置了访问控制规则，则仅有规则中所允许的访问才能进行。

相关配置

设置 NTP 的访问控制权限

- 可选配置
- 通过 `ntp access-group` 配置 NTP 访问控制权限及对应的 ACL

检验方法

通过 `show run` 查看 NTP 配置是否正确配置

相关命令

配置 NTP 服务的访问控制权限

- 【命令格式】 `ntp access-group { peer | serve | serve-only | query-only } access-list-number | access-list-name`
- 【参数说明】 **peer**：既允许对本地 NTP 服务进行时间请求和控制查询，也允许本地设备与远程系统同步时间（完全访问权限）。
- serve**：允许对本地 NTP 服务进行时间请求和控制查询，但不允许本地设备与远程系统同步时间。
- serve-only**：仅允许对本地 NTP 服务进行时间请求。
- query-only**：仅允许对本地 NTP 服务进行控制查询。
- access-list-number**：IP 访问控制列表标号；范围为 1 ~ 99 和 1300 ~ 1999。关于如何创建 IP 访问控制列表请参考《ACL》中的相关描述。
- access-list-name**：IP 访问控制列表名。关于如何创建 IP 访问控制列表请参考《访问控制列表配置指南》中的相关描述。
- 【命令模式】 全局模式
- 【使用指导】 配置 NTP 访问控制权限。
- 当一个访问请求到达时，NTP 服务按照从最小访问限制到最大访问限制的顺序依次匹配规则，以第一个匹配到的规则为准。匹配顺序为 peer、serve、serve-only、query-only。

配置举例

📌 NTP 访问控制权限配置

【配置方法】 配置只允许 192.168.1.1 的设备对本地设备进行时间同步请求


```
Ruijie(config)# access-list 1 permit 192.168.1.1
Ruijie(config)# ntp access-group serve-only 1
```

4.6 监视与维护

查看运行情况

作用	命令
show ntp status	显示当前的 NTP 信息

查看调试信息

 输出调试信息，会占用系统资源。使用完毕后，请立即关闭调试开关。

作用	命令
debug ntp	打开调试功能。
no debug ntp	关闭调试功能。

5 SNTP

5.1 概述

SNTP (Simple Network Time Protocol , 简单网络时间协议) 是 NTP 的简化版本 , 主要用来同步因特网中的计算机时钟。SNTP 适用于无需完全使用 NTP 功能的情况。

NTP 算法复杂 , 对系统要求较高。而 SNTP 在实现时 , 计算时间用了简单的算法 , 性能较高。而精确度一般也能达到 1 秒左右 , 也能基本满足绝大多数场合的需要。由于 SNTP 的报文和 NTP 的报文是完全一致的 , 所以设备实现的 SNTP Client 能完全兼容 NTP Server。

 下文仅介绍 SNTP 的相关内容。

协议规范

- RFC 2030 : Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI

5.2 典型应用

典型应用	场景描述
从 NTP 服务器同步时间	设备作为客户端 , 从 NTP 服务器同步时间

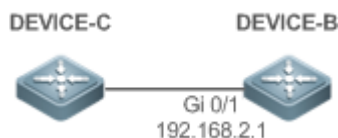
5.2.1 从 NTP 服务器同步时间

应用场景

如图所示 , DEVICE-B 将本地时钟作为 NTP 参考时钟源 , 向 DEVICE-C 提供时间同步服务。

DEVICE-C 作为 SNTP 客户端 , 从 DEVICE-B 同步时间。

图 4-1



功能部署

- 指定 DEVICE-B 为 DEVICE-C 的 SNTP 服务器。

- DEVICE-C 开启 SNTP 功能

5.3 功能详解

基本概念

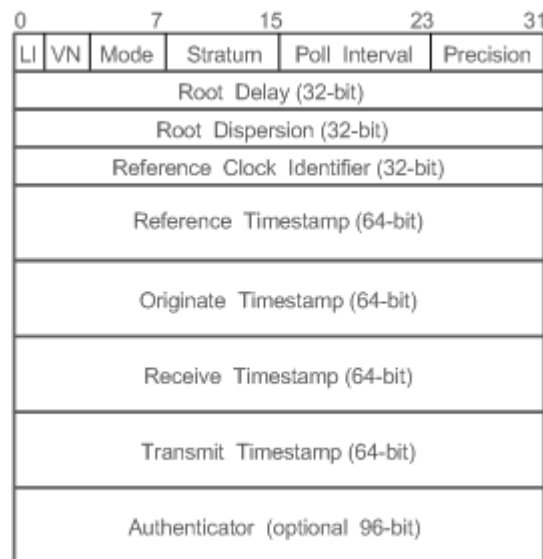
SNTP 报文

SNTPV4 是从 NTP 发展过来的，主要是简化 NTP 的功能。SNTPV4 并没有改变 NTP 规范和原有实现过程。SNTPV4 的消息格式于 RFC1305 中定义的 NTP 格式是一致的，只是一些数据域被初始化为预定的值。

同 RFC1305 定义，SNTP 采用 UDP 报文进行传输，UDP 端口号为 123。

NTP 时间同步报文格式如图 1-2

图 4-2 SNTP 时间同步报文格式



- Leap Indicator (LI): 2 比特，闰秒标志。

i 00-无警告信息 01-上一分钟有 61 秒 10-上一分钟有 59 秒 11-时钟未同步

- Version Number (VN): 3 比特，NTP/SNTP 版本号，当前版本号为 3。

- Mode : 3 比特，SNTP/NTP 工作模式。

i 0-未定义 1-主动对等体 2-被动对等体 3-客户端 4-服务器 5-广播 6-控制信息 7-保留

- Stratum : 8 比特，本地时钟的层数 (0-未定义 1-主参考时钟源 其它值-次参考时钟源)。

- Poll Interval : 8 位整数，轮询时间 (秒数)

- Precision : 8 位整数，本地时钟的时间精度 (秒数)

- Root Delay : 32 位整数，到主参考时钟源的往返时间

- Root Dispersion：32 位整数，相对于参考时钟源的最大误差
- Reference Clock Identifier：32 比特，参考时钟源的标识
- Reference Timestamp：64 位时间戳，最后一次被设置或者被校正的时间
- Originate Timestamp：64 位时间戳，时间同步请求报文离开客户端的本地时间
- Receive Timestamp：64 位时间戳，时间同步请求报文到达服务器的本地时间
- Transmit Timestamp：64 位时间戳，时间同步响应报文离开服务器的本地时间
- Authenticator（可选）：验证信息

功能特性

功能特性	作用
SNTP 时间同步	从 SNTP/NTP 服务器同步时间到本地设备。

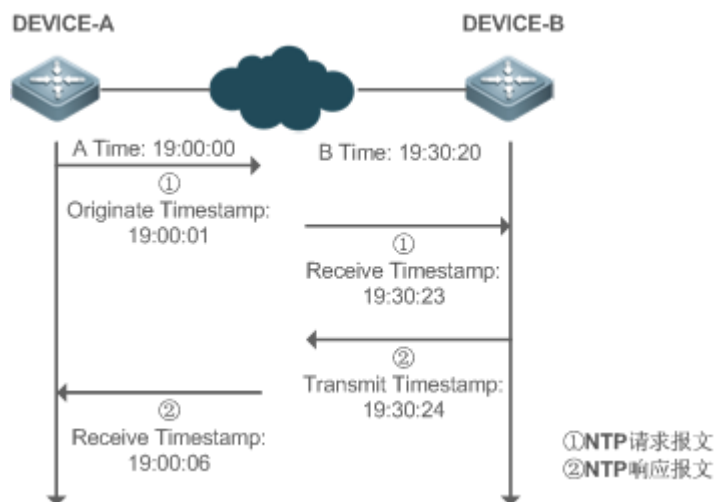
5.3.1 SNTP 时间同步

工作原理

SNTP 同步时间的方式是与服务器之间交互 SNTP/NTP 报文。客户端每隔一段时间（默认是半小时）向服务器发送时间同步报文。收到服务器响应报文后进行时间同步。

SNTP 时间同步报文格式如图 1-3

图 4-3 SNTP 基本工作原理图



DEVICE-B（下面简称 B）作为 NTP 参考时钟源，DEVICE-A（下面简称 A）作为 SNTP 客户端从 DEVICE-B 同步时间，在某一时刻 A 的本地时钟为 19:00:00，B 的本地时钟为 19:30:20：

8. A 发出 SNTP/NTP 请求报文，报文离开 A 的本地时间 (T0) 为 19:00:00，填充在 Originate Timestamp
9. 经过 2 秒的网络延时，B 收到请求报文的本地时间 (T1) 为 19:30:23，填充在 Receive Timestamp
10. B 处理 NTP 请求，1 秒后响应 NTP 报文，报文离开 B 的本地时间 (T2) 为 19:30:24，填充在 Transmit Timestamp
11. 经过 2 秒的网络延时，A 接收到响应报文，响应报文到达 A 的本地时间 (T3) 为 19:00:06

时间同步的具体算法如下：

- A 通过公式 $((T1-T0)+(T2-T3))/2$ 计算出 B 和 A 的时间差为 30 分 20 秒
- A 通过公式 $(T3-T0)-(T2-T1)$ 计算出 A 和 B 的报文往返的延时为 4 秒

相关配置

打开 SNTP

- 缺省 SNTP 状态是关闭的。
- 通过 **sntp enable** 命令开启 SNTP 功能

配置 SNTP 服务器

- 缺省情况下，没有配置 SNTP 服务器。
- 通过 **sntp server** 命令指定 SNTP 服务器。

配置 SNTP 同步时钟间隔

- 缺省情况下，SNTP 同步时钟的间隔是 1800s。
- 通过 **sntp interval** 命令指定 SNTP 服务器。

5.4 配置详解

配置项	配置建议&相关命令	
配置 SNTP	 必须配置，用于开启 SNTP 功能	
	sntp enable	打开 SNTP
	sntp server	配置 SNTP Server 的地址
	 可选配置，用于调整 SNTP 时间同步间隔	
	sntp interval	配置 SNTP 同步时钟的间隔

5.4.1 配置 SNTP

配置效果

SNTP Client 一定的时间间隔定期访问 NTP Server，可以定时校正时钟。

注意事项

通过 SNTP 协议通讯后获取的时间都是格林威治标准时间 (GMT)，为了准确的获取本地时间，需要设置本地时区来对标准时间进行调正。

配置方法

打开 SNTP

- 必须配置，缺省 SNTP 状态是 Disable。

配置 SNTP Server 的地址

- 必须配置，缺省没有设置 SNTP/NTP 服务器

配置 SNTP 同步时钟的间隔

- 可选配置
- 默认情况下，设备每隔半小时同步一次时间

检验方法

使用 **show sntp** 命令查看 SNTP 相关参数。

相关命令

打开 SNTP

【命令格式】 **sntp enable**

【参数说明】 -

【命令模式】 全局配置模式

【使用指导】 缺省 SNTP 状态是 Disable。

no sntp enable 全局配置命令来关闭 SNTP。

配置 SNTP Server 的地址

【命令格式】 **sntp server [oob] { ip- address | domain }**

【参数说明】 *ip-address* : SNTP 服务器的 IP 地址。缺省没有设置任何 SNTP 服务器。

domain: SNTP 服务器的域名。缺省没有设置任何 SNTP 服务器。

oob : SNTP 服务器支持带外管理接口 (interface of mgmt) 。

【命令模式】 全局配置模式

【使用指导】 由于 SNTP 协议和 NTP 完全兼容，所以这个 Server 完全可以配置成 internet 上公用的 NTP Server。

由于 SNTP 的报文和 NTP 的报文是完全一致的，所以 SNTP Client 能完全兼容 NTP Server。网络上存在着

较多的 NTP Server，用户可以选择一个网络延迟较少的一个作为设备上的 SNTP Server。

配置 SNTP 同步时钟的间隔

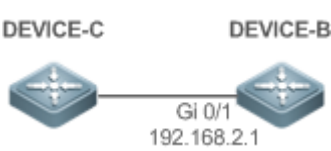
- 【命令格式】 `sntp interval seconds`
- 【参数说明】 `seconds`：定时同步的间隔，单位为“秒” 范围为 60 秒--65535 秒。缺省值为 1800s。
- 【命令模式】 全局配置模式
- 【使用指导】 该命令设置 SNTP Client 需要定时和 NTP/SNTP Server 同步时钟的时间间隔。

⚠ 这里设置的时间间隔不会立即生效，如果要立即生效，请配置完时间间隔后执行 `sntp enable` 命令。

配置举例

SNTP 时间同步

- 【网络环境】
图 4-4



- DEVICE-B：网络上的 NTP 服务器
- DEVICE-C：从 DEVICE-B 同步时间

- 【配置方法】 DEVICE-C 开启 SNTP 功能，NTP 服务器配置为 DEVICE-B

DEVICE-C

```
C#configure terminal
C(config)# sntp server 192.168.2.1
C(config)# sntp enable
C(config)# exit
```

- 【检验方法】
 - 在 DEVICE-C 上通过 `show clock` 命令可以查看时间是否成功同步。
 - 在 DEVICE-C 上 `show sntp` 查看 sntp 状态和服务器是否配置成功

5.5 监视与维护

清除各类信息

查看运行情况

作用	命令
show sntp	查看 SNTP 的相关参数

查看调试信息

 输出调试信息，会占用系统资源。使用完毕后，请立即关闭调试开关。

作用	命令
debug sntp	打开调试功能。

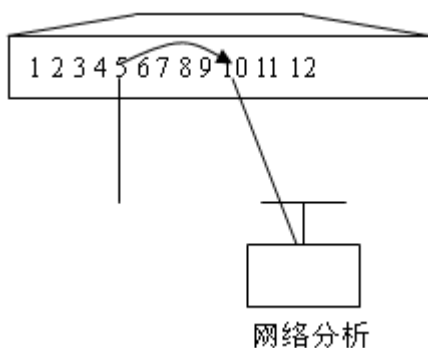
6 SPAN-RSPAN

6.1 概述

镜像(SPAN)是将指定端口的报文复制到交换机上另一个连接有网络监测设备的端口，进行网络监控与故障排除。

通过 SPAN 可以监控所有进入和从源端口输出的报文。例如，在下图中，端口 5 上的所有报文都被映射到了端口 10，连接在端口 10 上的网络分析仪虽然没有和端口 5 直接相连，但是可以接收通过端口 5 上的所有报文。

图 5-1 SPAN 配置实例



镜像功能主要应用于在网络监控和故障排查两种场景中，用于对网络信息的监控和网络故障的解决。

RSPAN(Remote SPAN，远程镜像)是 SPAN 的扩展，能够远程监控多台设备，每个 RSPAN 会话建立于用户指定的 Remote VLAN 内。远程镜像突破了被镜像端口和镜像端口必须在同一台设备上的限制，使被镜像端口和镜像端口间可以跨越多个网络设备，这样用户就可以坐在中心机房通过分析仪观测远端被镜像端口的数据报文了。

远程镜像的应用场景和本地镜像类似，但使得用户不必呆在机房就可以对数据进行实时监控，极大地方便用户。

VSPAN 是 VLAN SPAN 的简称，是指将某些 VLAN 的数据流作为数据源镜像到目的端口，它和基于端口的镜像配置方式类似。VSPAN 具有以下特性：

- 可以指定某个 VLAN 作为镜像的数据源，这个 VLAN 不能是 Remote VLAN。
- 可以指定某些 VLAN 作为镜像的数据源，这些 VLAN 不能是 Remote VLAN。
- 配置 VLAN 做为源时只能基于 rx 方向的报文镜像。

协议规范

- 无

6.2 典型应用

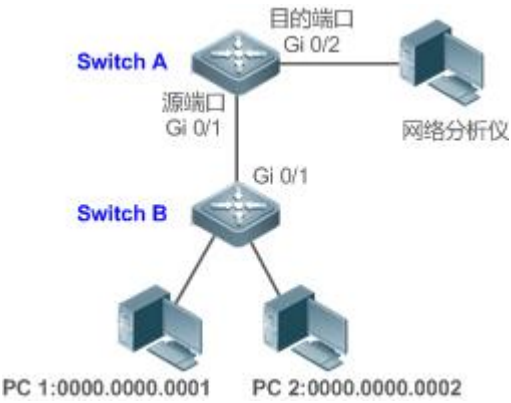
典型应用	场景描述
基于流的镜像	需要监控具有特定特征的数据流，比如监控指定 ACL 策略的数据流。
一对多的镜像	需要多个用户对同一端口的数据进行监控。
RSPAN 基本应用	需要将镜像源设备的报文镜像到目的的设备上进行监控。
基于 AP 口的双发应用	去堆叠场景下管理口解决方案。

6.2.1 基于流的镜像

应用场景

如图所示，通过适当的配置，网络分析仪能够监控 Switch A 转发给 Switch B 的所有数据流，监控来自 Switch B 的特定数据流（如来自 PC1 和 PC2 的数据流）。

图 5-2 SPAN 简单应用拓扑



【注释】 0000.0000.0001 为 PC1 的 MAC 地址。
0000.0000.0002 为 PC2 的 MAC 地址。

功能部署

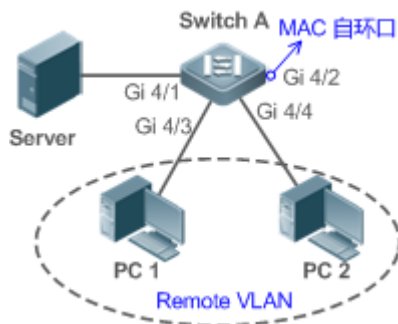
- 上图中，在连接网络分析仪的设备 Switch A 上配置 SPAN 功能，将连接 Switch B 的端口 Gi 0/1 设置为 SPAN 的源端口，将直连网络分析仪的端口 Gi 0/2 设置为 SPAN 的目的端口。
- 配置 SPAN 源端口 Gi 0/1 基于流的镜像（仅允许 PC1 和 PC2 的数据流）。

6.2.2 一对多的镜像

应用场景

如图所示,在单台设备上实现一对多镜像,即 PC1 和 PC 2 均可监控服务器相连端口的收发流量。用户可以通过适当的配置(远程 VLAN、MAC 自环口等), 可以在 PC1、PC 2 中对流经 Gi 4/1 的数据流进行监控, 从而实现对服务器数据流的监控。

图 5-3 一对多镜像应用拓扑



【注释】 Remote VLAN : 远程 VLAN。

功能部署

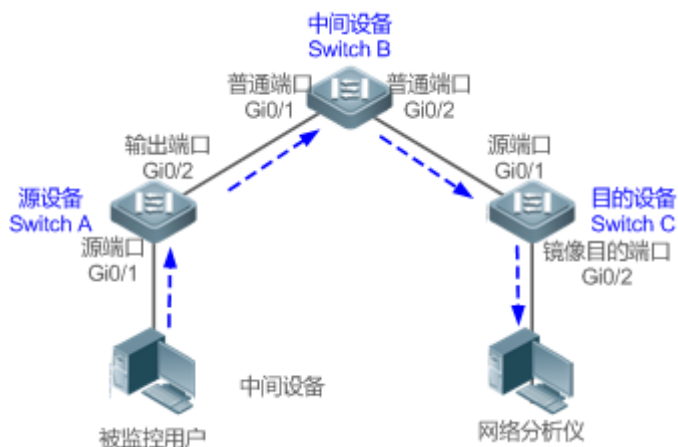
- 在设备 Switch A 上创建 Remote VLAN。
- 指定设备 Switch A 为 RSPAN 的源设备, 配置直连服务器的端口 Gi4/1 为镜像源端口; 选择一个 Down 状态的端口本例为 Gi 4/2 为镜像输出端口, 将该端口加入 Remote VLAN, 并配置 MAC 自环 (可以在接口模式下通过 **mac-loopback** 命令进行配置)。
- 将直连 PC1 和 PC2 的端口加入 Remote VLAN。

6.2.3 RSPAN 基本应用

应用场景

如图所示,网络分析仪可以通过远程镜像功能,实现在目的设备 Switch C 上通过中间设备 Switch B 监控连接到源设备 Switch A 上的用户。且设备之间均能正常交换数据。

图 5-4 RSPAN 基本应用拓扑



【注释】 -

功能部属

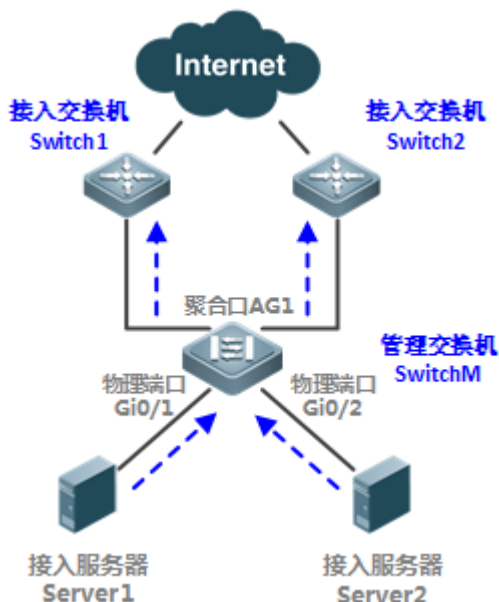
- 在源设备 Switch A、中间设备 Switch B 和目的设备 Switch C 上配置 Remote VLAN。
- 在源设备上，配置直连用户的端口 Gi 0/1 为源端口，与中间设备相连的端口 Gi 0/2 为输出端口，并配置输出端口可交换功能。
- 在中间设备上，与源设备、目的设备相连的端口 Gi 0/1 和 Gi 0/2 仅需配置为普通端口。
- 在目的设备上，与中间设备相连的端口 Gi 0/1 作为源端口，仅需配置为普通端口，与网络分析仪相连的端口 Gi 0/2 配置为镜像目的端口，并配置镜像目的端口可交换功能。

6.2.4 去堆叠场景下的管理口解决方案

应用场景

如图所示，在去堆叠场景下，服务器的 iLO 口通过管理交换机的 AP 口双上联到一组(两台)接入交换机，为了能够使两台接入交换机同时学习到服务器 iLO 口的 ARP 或者 ND，可以通过在管理交换机上开启 AP 口双发功能，实现上联接入交换机的 AP 口双发 ARP、ND 协议报文。

图 6-1 去堆叠场景拓扑简图



【注释】 在管理交换机上，物理口 Gi0/1 和 Gi0/2 作为下联口连接接入服务器的 iLO 口，物理口 te0/49,50 中的两个作为聚合口 AG1 的成员上联接入交换机 Switch1 和 Switch2 设备

功能部属

- 管理交换机的下联口仅需配置二层 VLAN，然后配置 te0/49 和 te0/50 为 LACP 聚合口成员。
- 两台接入交换机上，分别配置对应的物理口为 LACP 聚合口成员，同时利用 LACP 的 sysmac 可配置功能，将 LACP 的 sysmac 配置为相同的 mac，使其在逻辑上变成同一个 LACP。
- 在管理交换机上指定上联的 AP 口开启双发功能，实现下联所有服务器的指定类型(ARP 和 ND)报文能够同时发送到两台接入交换机上。配置命令如下：

```
Ruijie(config)#packet double-distribute arp_nd interface aggregatePort 1
```

6.3 功能详解

基本概念

SPAN 会话

SPAN 会话是镜像源端口与目的端口之间的数据流，可以监控单个或多个端口的输入、输出、双向的报文。Switched Port、Routed Port 和 AP(聚合端口)等类型的端口都可以配置为 SPAN 会话的源端口和目的端口。端口加入 SPAN 会话后并不影响交换机的正常操作。

用户可以在处于关闭状态的端口上配置 SPAN 会话，但是该 SPAN 会话是非活动的，只有相关的端口被打开后，SPAN 会话才会变为活动状态。另外，SPAN 会话在交换机上电后并不立即生效，直到目的端口处于可操作状态后，SPAN 会话才处于活动状态。用户可以通过 **show monitor [session session-num]** 命令查看 SPAN 会话的操作状态。

📌 镜像数据流

SPAN 会话包含以下三种方向的数据流：

- 输入数据流：所有源端口上接收到的报文都将被复制一份到目的端口。在一个 SPAN 会话中，用户可以监控一个或多个源端口的输入报文。由于某些原因(如端口安全)，从源端口输入的报文可能被丢弃，但这不影响 SPAN 功能，该报文仍然会镜像到目的端口。
- 输出数据流：所有从源端口发送的报文都将复制一份到目的端口。在一个 SPAN 会话中，用户可以监控一个或多个源端口的输出报文。若由于某些原因，从别的端口发送到源端口的报文可能被丢弃，同样，该报文也不会发送到目的端口。由于某些原因从源端口输出的报文的格式可能改变，例如源端口输出经过路由之后的报文，报文的源 MAC、目的 MAC、VLAN ID 以及 TTL 发生变化，同样，拷贝到目的端口的报文的格式也会变化。
- 双向数据流：包括上面所说的两种数据流。在一个 SPAN 会话中，用户可监控一个或多个源端口的输入和输出方向的数据流。

📌 源端口

源端口也被称为被监控口，在 SPAN 会话中，源端口上的数据流被监控，用于网络分析或故障排除。在单个 SPAN 会话中，用户可以监控输入、输出和双向数据流，且源端口的最大个数没有限制。

源端口具有以下特性：

- 源端口可以是 Switched Port、Routed Port 或 AP。
- 源端口不能同时作为目的端口。
- 源端口和目的端口可以属于同一 VLAN，也可以属于不同 VLAN。

📌 目的端口

SPAN 会话有一个目的端口(也被称为监控口)，用于接收源端口的报文拷贝。

目的端口具有以下特性：

- 目的端口可以是 Switched Port、Routed Port 或 AP。
- 目的端口不能同时作为源端口。

功能特性

功能特性	作用
SPAN	同一设备上端口的镜像。
RSPAN	跨设备的端口镜像。

6.3.1 SPAN

本地镜像主要是用来监控交换机上的数据流。通过将一个端口上的帧拷贝到交换机上另一个连接有网络分析设备或 RMON 分析仪的端口上来分析该端口上的通讯。

工作原理

端口收发报文时检测用户如果有配置该端口作为镜像源时，则会将该端口收发的报文复制到目的端口一份。

配置镜像源端口

用户需要指定镜像会话 ID、源端口名字来配置镜像源端口，并通过镜像方向的可选配置项决定镜像数据流的方向或通过指定 ACL 策略镜像特定数据流。

配置镜像目的端口

用户需要指定镜像会话 ID、目的端口名字来配置镜像目的端口，并通过交换功能可选配置项决定是否在该目的镜像端口上开启交换功能和剥离 TAG 信息功能。

相关配置

系统镜像功能默认是关闭的，只有用户创建会话，并配置源和目的镜像端口才会开启镜像功能。镜像会话可以在配置镜像的源端口或者目的端口的时候进行创建。

配置镜像源端口

缺省情况下，镜像会话中没有镜像源端口。用户通过下面命令配置镜像源端口。

```
monitor session session-num source interface interface-id [ both | rx | tx ] [ acl name ]
```

其中，

session-num：镜像会话 ID，针对不同产品支持镜像会话个数会有所不同。

interface-id：待配置的镜像源端口。

rx：配置 **rx** 选项后，只监听源端口接收的报文。

tx：配置 **tx** 选项后，只监听源端口发送的报文。

both：配置 **both** 选项后，源端口收发的报文都会送到目的端口进行监听，即包含 **rx** 和 **tx**。如果用户不配置 **rx**、**tx** 和 **both** 三个选项中的任何一个则默认开启 **both** 选项。

acl：配置该选项时则需要用户指定一个 ACL 策略，即监听源端口上该策略允许的报文，默认不开启该功能。

配置镜像目的端口

缺省情况下，镜像会话中没有镜像源端口。用户通过下面命令配置镜像的目的端口。

```
monitor session session-num destination interface interface-id [switch ]
```


其中，

switch：在配置镜像目的口时，如果没有打开该选项，则镜像目的口只接收镜像源的镜像报文，其它报文均丢弃。如果打开该选项，除了接收镜像源的镜像报文同时非源端口送过来的报文也不会丢弃，即不影响目的口和外界的其他通信。

配置镜像目的端口时，如果没有配置 **switch** 选项则默认关闭相应功能。

配置基于流的镜像

缺省情况下，该功能关闭。用户通过 **monitor session session-num source interface interface-id [rx] acl acl-name** 命令配置基于流的镜像。

 使用过程中，用户需要特别注意以下几点：

- 镜像的目的端口参与 STP 树的计算。
- 如果改变了源端口的 VLAN 配置，配置将马上生效。
- 如果改变了目的端口的 VLAN 配置，配置马上生效。
- 如果禁用了源端口或目的端口，SPAN 将不起作用。
- 如果将源端口或目的端口加入 AP，源端口或目的端口将退出 SPAN 会话。
- 如果 VLAN（VLAN 列表）做为镜像源时，要保证目的口有足够大的带宽能够接收整个 VLAN 的镜像数据。
- 产品的差异性，并不是所有产品都支持上述命令的所有选项。

6.3.2 RSPAN

RSPAN 能够远程监控多台设备，每个 RSPAN Session 建立于用户指定的 Remote VLAN 内。远程镜像突破了被镜像端口和镜像端口必须在同一台设备上的限制，使被镜像端口和镜像端口间可以跨越多个网络设备。

工作原理

远程镜像的原理是原设备、中间设备和目的设备通过创建一个 Remote VLAN，且所有参与会话的端口都要加入该 Remote VLAN 中，镜像报文在 Remote VLAN 内进行广播，使得镜像报文从源交换机的源端口传送到目的交换机的目的端口。

配置远程 VLAN

镜像源端口的报文就是通过在远程 VLAN 进行广播来实现报文从本台交换机复制到远程交换机的。镜像源端口、输出端口、反射端口及中间设备的透传端口（中间设备的报文进入端口、输出端口）和目的交换机的目的端口及进入端口都必需位于该远程 VLAN 内。该功能需要在 VLAN 模式下将 VLAN 配置远程 VLAN。

配置远程镜像会话

远程镜像源端口和目的端口的配置和本地镜像类似，但是在配置时指定的镜像会话 ID 必需是远程镜像。

配置远程镜像源端口

配置远程镜像源端口和配置本地镜像源端口一样，只是在指定镜像会话 ID 时，需要使用远程镜像会话 ID。

配置远程镜像输出端口

输出端口和反射口均位于源设备中。如果用户希望远程镜像能够实现一对多镜像时，则需要配置反射口来实现。如果用户配置的远程镜像实现的仍是一对一的镜像，则不需要配置反射口只需配置一个输出端口即可。

反射口和输出端口都必需位于远程 VLAN 内，源端口被镜像的报文在该远程 VLAN 内进行广播。源设备中就是通过反射端口、输出端口将报文传送到中间交换机或目的交换机中。

配置远程镜像目的端口

配置远程镜像的目的端口时必需指定远程镜像会话 ID，远程 VLAN 及端口名字，这样源端口的报文就可以通过远程 VLAN 将报文从源端口复制到目的端口。

配置基于流的远程镜像

RSPAN 是对本地 SPAN 的扩展，因此 RSPAN 同样也支持基于流的镜像，具体配置同基于流的 SPAN 配置。基于流的 RSPAN 不影响正常通讯。

用户可以在 RSPAN 源设备上配置源端口的 in 方向的 ACL，支持标准 ACL、扩展 ACL、MAC ACL、自定义 ACL。

用户可以在 RSPAN 源设备上配置源端口的 in 方向的端口 ACL，可以在 RSPAN 目的设备上配置目的端口 out 方向的端口 ACL。用户可以在 RSPAN 源交换机上基于 Remote VLAN 应用 out 方向的 ACL，在 RSPAN 目的交换机上基于 Remote VLAN 应用 in 方向的 ACL。

配置一对多的镜像

如果用户需要将同一源端口的数据流镜像到多个目的端口，可以配置一个 RSPAN 会话，该 RSPAN 的源口为一对多镜像源端口，转发口(即为源设备的输出端口)为非一个多镜像目的的其它以太网口。同时在 RSPAN 转发口的接口模式下配置 MAC 自环功能。注意该 RSPAN 会话中的所有关联端口均需要加入到远程 VLAN 中。

相关配置

远程镜像功能默认是关闭的，只有用户创建远程镜像会话，并配置远程 VLAN、源和目的镜像端口才会开启该功能。

配置远程 VLAN

缺省情况下，RSPAN 没有指定远程 VLAN。用户可以在 VLAN 模式下，通过 **remote-span** 命令将该 VLAN 配置为远程 VLAN。一个远程 VLAN 对应一个 RSPAN 会话。

配置远程镜像的源设备

缺省该功能关闭。用户可以在全局模式下通过 **monitor session session-num remote-source** 命令将该设备配置为指定 RSPAN 会话的远程源设备。

配置远程镜像的目的设备

缺省该功能关闭。用户可以在全局模式下通过 **monitor session session-num remote-destination** 命令将该设备配置为指定 RSPAN 会话的远程目的设备。

配置远程镜像源端口


源设备配置会话的源端口，和本地镜像配置源端口一样，只是要用远程会话 ID。缺省该功能关闭。

配置远程源镜像的输出目的端口

缺省该功能关闭。用户可以在全局模式下通过 **monitor session session-num destination remote vlan remote-vlan interface interface-name [switch]** 命令配置源镜像的输出目的端口。可选项 **switch** 配置的情况下，表示该输出目的口可以参与正常的数据报文交换，缺省该可选项是不配置的。注意输出端口必须加入到远程 VLAN 中。

配置远程目的设备的目的口

缺省该功能关闭。用户可以在全局模式下通过 **monitor session session-num destination remote vlan remote-vlan interface interface-name [switch]** 配置远程目的设备的目的端口。可选项 **switch** 配置的情况下，表示该输出目的口可以参与正常的数据报文交换，缺省该可选项是不配置的。注意目的端口必须加入到远程 VLAN 中。

 使用过程中，用户需要特别注意以下几点：

- Remote VLAN 必需在每台设备中都要进行配置，且 VLAN ID 必须一致，并且所有参与会话的端口都要加入该 VLAN 中。
- 建议不要将普通端口加入 Remote VLAN。
- 不要在与中间交换机或目的交换机相连的端口上配置镜像源端口，否则可能引起网络内的流量混乱。

6.3.3 AP 口双发功能

在去堆叠场景下，服务器的 iLO 口通过管理交换机的 AP 口双上联到一组接入交换机，为了能够使两台接入交换机同时学习到服务器 iLO 口的 ARP 或者 ND，可以通过在管理交换机上开启 AP 口双发功能，实现上联接入交换机的 AP 口双发 ARP、ND 协议报文。

工作原理

AP 口双发的原理是使用镜像功能，使得从管理交换机的下联物理口发送到上联 AP 口的特定协议报文，会转发到 AP 的每个成员端口，AP 成员最多支持 2 个，因此称为 AP 口双发功能。

配置 AP 口双发功能


配置 AP 口双发的功能仅需指定需要双发的协议报文和双发的 AP 端口。镜像的处理由管理交换机内部实现。

相关配置

AP 口双发功能默认是关闭的，只有配置指定协议报文应用到 AP 端口上才会开启该功能。

配置 AP 口双发功能

缺省该功能关闭。用户可以在全局模式下通过 **packet double-distribute {arp | nd | arp_nd} interface interface-id** 命令指定 AP 口开启双发功能。

 使用过程中，用户需要特别注意以下几点：

- AP 口的成员数不能超过 2 个。

- ARP 报文和 ND 报文可以单独配置，也可以叠加配置。分别配置 arp 和 nd 后，会自动叠加为 arp_nd。
- 该命令仅支持应用在一个 Aggregate Port 口上，重复配置会以最后一次配置为准。

6.4 产品说明



S6000E 上，源口 tx 镜像能同时镜像 CPU 发出的报文。

6.5 配置详解

配置项	配置建议 & 相关命令	
配置 SPAN 基本功能	⚠ 必须配置。用于创建本地镜像。	
	monitor session <i>session-num</i> source interface <i>interface-id</i> [both rx tx]	配置镜像源端口
	monitor session <i>session-num</i> destination interface <i>interface-id</i> [switch]	配置镜像目的端口
	monitor session <i>session-num</i> source interface <i>interface-id</i> rx acl <i>acl-name</i>	配置基于流的镜像
	monitor session <i>session-num</i> source vlan <i>vlan-id</i> [rx]	指定某个 VLAN 作为镜像的数据源
	monitor session <i>session-num</i> source filter vlan <i>vlan-id-list</i>	指定某些 VLAN 作为镜像的数据源
配置 RSPAN 基本功能	⚠ 必须配置。用于创建远程镜像。	
	monitor session <i>session-num</i> remote-source	配置远程镜像会话 ID 并指定为源设备
	monitor session <i>session-num</i> remote-destination	配置远程镜像会话 ID 并指定为目的设备
	remote-span	配置远程 VLAN
	monitor session <i>session-num</i> source interface <i>interface-id</i> [both rx tx]	配置远程源镜像源端口
	monitor session <i>session-num</i> destination remote vlan <i>remote-vlan-id</i> interface <i>interface-id</i> [switch]	配置远程源镜像的输出端口或者远程目的镜像的目的端口
配置 AP 口双发功能	packet double-distribute { arp nd arp_nd } interface <i>interface-id</i>	指定 AP 口开启双发功能

6.5.1 配置 SPAN 基本功能

配置效果

- 配置镜像会话的源和目的端口。
- 目的口可以监控到任何进出源端口的报文。

注意事项

- 如果将源端口或目的端口加入 AP，源端口或目的端口将退出 SPAN 会话。
- 如果镜像目的口没有开启 switch 功能，则目的口只能接收镜像报文，其它流经该端口的报文将被丢弃。开启后可以接收非镜像报文。

配置方法

📌 镜像会话

- 全局模式。必须配置。
- 可以通过配置镜像的源端口或者目的端口时同时配置镜像会话。还可以通过配置指定某个 VLAN 或者某些 VLAN 作为镜像的数据源时配置镜像会话。

📌 配置镜像源端口

- 全局模式。必须配置。
- 配置镜像源端口时可以选择配置的镜像方向，缺省是 both 方向，即同时监测报文的接收和发送行为。

📌 配置镜像目的端口

全局模式。必须配置。

只有同时配置镜像的源端口或者指定 VLAN 作为镜像数据源，以及配置镜像的目的端口时，该镜像会话才真正起作用。

检验方法

- 镜像配置的校验也可以通过 **show monitor** 或者 **show running** 命令查看。也可以在镜像目的的口上进行抓包分析，通过抓取的报文查看镜像功能是否生效。

相关命令

📌 配置镜像源端口

【命令格式】 **monitor session session-num source interface interface-id [both | rx | tx]**

【参数说明】 **session-num**：镜像会话 ID

interface-id：接口名字

both：同时监控输入和输出方向的报文，为缺省值

rx：监控输入方向的报文

tx：监控输出方向的报文

【命令模式】 全局模式

【使用指导】 -

配置镜像目的端口

【命令格式】 **monitor session session-num destination interface interface-id [switch]**

【参数说明】 *session-num* : 镜像会话 ID

interface-id : 接口名字

switch : 支持镜像目的口交换功能，缺省为不打开

【命令模式】 全局模式

【使用指导】 -

配置基于流的镜像

【命令格式】 **monitor session session-num source interface interface-id rx acl acl-name**

【参数说明】 *session-num* : 镜像会话 ID

interface-id : 接口名字

acl-name : acl 名字

【命令模式】 全局模式

【使用指导】 -

指定某个 VLAN 作为镜像的数据源

【命令格式】 **monitor session session-num source vlan vlan-id [rx]**

【参数说明】 *session-num* : 镜像会话 ID

vlan-id : 指定的 VLAN ID

rx : 监控输入方向的报文

【命令模式】 全局模式

【使用指导】 -

指定某些 VLAN 作为镜像的数据源

【命令格式】 **monitor session session-num source filter vlan vlan-id-list**


【参数说明】 *session-num* : 镜像会话 ID

vlan-id-list : 指定的某些 VLAN ID

【命令模式】 全局模式

【使用指导】 -

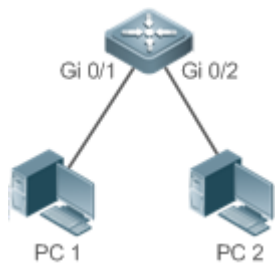
配置举例

 以下配置举例，仅介绍与本地镜像相关的配置。

下面以本地镜像为例介绍

【网络环境】

图 5-5



- 【配置方法】
- 如图 1-5，配置设备 A 的 Gi 0/1 和 Gi 0/2 属于 VLAN 1。
 - 创建 SVI 1，并配置 SVI 1 地址为 10.10.10.24。
 - 配置 PC1、PC2 地址为 10.10.10.1/24、10.10.10.2/24，略。
 - 配置设备 A 的本地镜像，指定端口 Gi 0/1 和 Gi 0/2 分别为镜像的源端口和目的端口。

A

```
Ruijie# configure
Ruijie(config)# vlan 1
Ruijie(config-vlan)# exit
Ruijie(config)# interface vlan 1
Ruijie(config-if-VLAN 1)# ip address 10.10.10.10 255.255.255.0
Ruijie(config-if-VLAN 1)# exit
Ruijie(config)# monitor session 1 source interface gigabitEthernet 0/1
Ruijie(config)# monitor session 1 destination interface gigabitEthernet 0/2
```

- 【检验方法】
- 首先通过 **show monitor 命令**查看镜像是否正确配置，配置成功后 PC1 向 SVI 1 发送 PING 包，PC2 利用抓包工具进行监控。

A

```
Ruijie# show monitor
sess-num: 1
span-type: LOCAL_SPAN
src-intf:
GigabitEthernet 0/1      frame-type Both
dest-intf:
GigabitEthernet 0/2
```

常见错误

- 用户配置镜像源端口和目的端口时指定的会话 ID 不一致。
- 带宽大的端口被镜像到带宽小的端口可能会造成丢包。

6.5.2 配置 RSPAN 基本功能

配置效果

- 配置远程镜像会话源设备中的源端口和输出端口或反射口，配置目的设备中的目的端口。
- 远程目的设备中的目的口可以监控到任何进出源端口的报文。

注意事项

- 如果将源端口或目的端口加入 AP，源端口或目的端口将退出 SPAN 会话。
- 如果远程镜像目的口没有开启 switch 功能，则目的口只能接收镜像报文，其它流经该端口的报文将被丢弃。开启后可以接收非镜像报文。
- 所有参与镜像的报文均要加入远程 VLAN 中。
- 中间设备必需创建远程 VLAN，且透传端口要加入该 VLAN。

配置方法

远程镜像会话

- 全局模式。必须配置。
- 需要在镜像源设备和镜像目的设备上配置相同的会话 ID，保证

配置源镜像设备

- 全局模式。必须配置。
- 用于指定被远程镜像监控的设备。

配置目的镜像设备

- 全局模式。必须配置。
- 用于指定远程镜像报文输出目的的设备。

配置远程镜像源端口

- 全局模式。必须配置。
- 在远程源镜像设备上配置。通过配置该功能，实现对远程镜像源端口的报文进行远程镜像监控。可以指定对经过该镜像源端口的输入方向、输出方向或者输入输出双向的 Remote VLAN 报文进行监控。

配置远程镜像输出端口

- 全局模式。必须配置。
- 在远程源镜像设备上配置。通过配置该功能，实现将 Remote VLAN 接收到的镜像报文通过输出端口输出到远程镜像目的设备上。实现一对多远程镜像时需要同时配置回环口和输出端口，实现一对一远程镜像时只需配置输出端口。

配置远程镜像目的端口

- 全局模式。必须配置。

- 在远程目的镜像设备上配置。通过配置该功能，远程目的设备将 Remote VLAN 接收到的镜像报文通过目的端口转发给监控设备。

检验方法

- 用户可以通过 **show monitor** 或者 **show running** 命令查看远程镜像中每台设备上面的配置是否成功。也可以在目的镜像设备上的目的镜像口抓包检查是否抓到了源镜像设备上的源端口镜像过来的报文。

相关命令

配置远程源镜像

- 【命令格式】 **monitor session session-num remote-source**
- 【参数说明】 *session-num*：远程镜像会话 ID
- 【命令模式】 全局模式
- 【使用指导】 -

配置远程目的镜像

- 【命令格式】 **monitor session session-num remote-destination**
- 【参数说明】 *session-num*：远程镜像会话 ID
- 【命令模式】 全局模式
- 【使用指导】 -

配置远程 VLAN

- 【命令格式】 **remote-span**
- 【参数说明】 -
- 【命令模式】 VLAN 模式
- 【使用指导】 -

配置源镜像源端口

- 【命令格式】 **monitor session session-num source interface interface-id [both | rx | tx] [acl acl-name]**
- 【参数说明】 *session-num*：镜像会话 ID
interface-id：接口名字
both：同时监控输入和输出方向的报文，为缺省值
rx：监控输入方向的报文
tx：监控输出方向的报文
acl-name：ACL 名字
- 【命令模式】 全局模式
- 【使用指导】 和本地镜像配置源端口一样，但是指定的会话 ID 为远程镜像。

配置远程源镜像输出端口、反射口

- 【命令格式】 **monitor session session-num destination remote vlan remote-vlan interface interface-id**
[**switch**]
- 【参数说明】
session-num : 镜像会话 ID
remote-vlan : 远程 VLAN
interface-id : 接口名字
switch : 是否参与报文交换
- 【命令模式】 全局模式
- 【使用指导】 实现一对多远程镜像时需要同时配置回环口和输出端口，实现一对一远程镜像时只需配置输出端口。

配置远程目的镜像的目的端口

- 【命令格式】 **monitor session session-num destination remote vlan remote-vlan interface interface-id**
[**switch**]
- 【参数说明】
session-num : 镜像会话 ID
remote-vlan : 远程 VLAN
interface-id : 接口名字
switch : 是否参与报文交换
- 【命令模式】 全局模式
- 【使用指导】 -

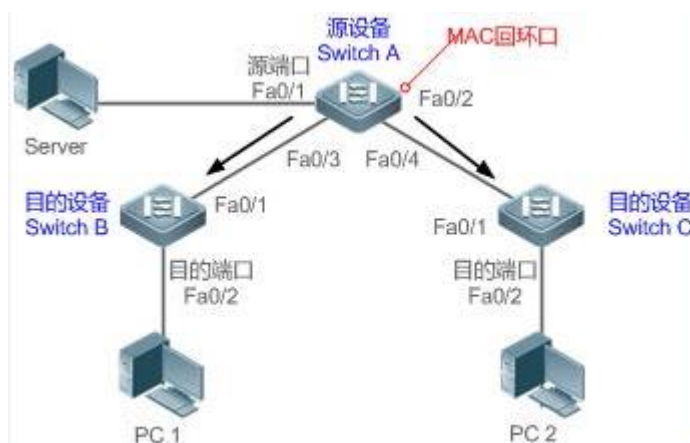
配置举例

以下配置举例，仅介绍与远程镜像（一对多）相关的配置。

下面以远程镜像为例介绍

【网络环境】

图 5-6



- 【配置方法】
- 如上图，设备 A、设备 B 和设备 C 配置远程 VLAN。
 - 在设备 A 中配置源端口、输出端口、回环口。
 - 在设备 B 和设备 C 中配置目的端口。

A

```
Ruijie# configure
Ruijie(config)# vlan 7
```

```
Ruijie(config-vlan)# remote-span
Ruijie(config-vlan)# exit
Ruijie(config)# monitor session 1 remote-source
Ruijie(config)# monitor session 1 source interface fa 0/1 both
Ruijie(config)# monitor session 1 destination remote vlan 7 interface fa 0/2 switch
Ruijie(config)# interface fa0/2
Ruijie(config-if)# mac-loopback
Ruijie(config-if)# switchport access vlan 7
Ruijie(config-if)# exit
Ruijie(config)# interface range fa0/3-4
Ruijie(config-if-range)# switchport mode trunk
```

B、C

```
Ruijie(config)# vlan 7
Ruijie(config-vlan)# remote-span
Ruijie(config-vlan)# exit
Ruijie(config)# monitor session 1 remote-destination
Ruijie(config)# monitor session 1 destination remote vlan 7 interface fa 0/2
Ruijie(config)# interface fa0/1
Ruijie(config-if)# switchport mode trunk
```

【检验方法】 分别在设备 A、设备 B 和设备 C 中执行 **show monitor** 或者 **show running** 命令查看镜像配置成功与否。

A

```
Ruijie# show monitor
sess-num: 1
span-type: SOURCE_SPAN
src-intf:
FastEthernet 0/1      frame-type Both
dest-intf:
FastEthernet 0/2
Remote vlan 7
mtp_switch on
```

B

```
Ruijie# show monitor
sess-num: 1
span-type: DEST_SPAN
dest-intf:
FastEthernet 0/2
Remote vlan 7
mtp_switch on
```

C

```
Ruijie# show monitor
sess-num: 1
span-type: DEST_SPAN
dest-intf:
FastEthernet 0/2
Remote vlan 7
```



```
mtp_switch on
```

常见错误

- 源设备、中间设备、目的设备均要配置远程 VLAN 且 VID 必须一致。
- 带宽大的端口被镜像到带宽小的端口可能会造成丢包。
- 如果实现一对多镜像时则需要配置一个反射口和若干输出端口。

6.5.3 配置 AP 口双发功能

配置效果

- 配置 AP 口的双发功能，两台接入交换机可以同时收到接入服务器的 ARP 和 ND 报文。

注意事项

- 仅需要在管理交换机上配置。
- 开启双发功能的 AP 成员口数量不能超过 2 个。

配置方法

📌 AP 口双发功能

- 全局模式。必须配置。
- 需要在管理交换机上配置，指定 AP 端口

检验方法

- 用户可以通过 **show running** 命令查看配置是否成功。也可以在两台接入交换机上查看对应的 ARP 表项或者 DN 表项。

相关命令

📌 配置远程源镜像

【命令格式】 **packet double-distribute {arp | nd | arp_nd} interface interface-id**

【参数说明】 **arp**：对 ARP 报文开启双发功能
nd：对 ND 报文开启双发功能
arp_nd：对 ARP 和 ND 报文都开启双发功能
interface-id：接口名字

- 【命令模式】 全局模式
- 【使用指导】 此命令用来配置通过指定的 Aggregate Port 口对下联口上报的报文进行双发。
该命令仅支持 Aggregate Port 口，且 Aggregate Port 中成员口数量不能超过 2 个。
该命令仅支持应用在一个 Aggregate Port 口上，重复配置会以最后一次配置为准。

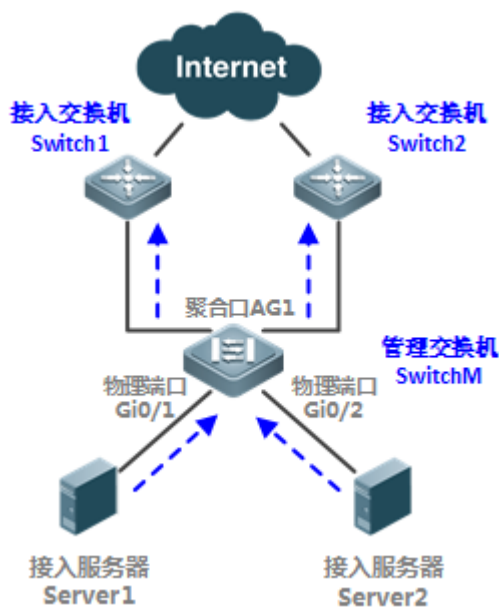
配置举例

以下配置举例，仅介绍与本地镜像相关的配置。

下面以本地镜像为例介绍

【网络环境】

图 6-8



- 【配置方法】
- 如图 1-8，配置管理交换机的 Gi 0/1 和 Gi 0/2 属于 VLAN 1。
 - 创建 SVI 1，并配置 SVI 1 地址为 10.10.10.10/24。
 - 配置 PC1、PC2 地址为 10.10.10.1/24、10.10.10.2/24，略。
 - 配置管理交换机与接入交换机相连的上联口加入聚合组 AG1，略。
 - 配置管理交换机在 AG1 上开启双发功能，双发 arp 报文。

```
Ruijie# configure
Ruijie(config)# vlan 1
Ruijie(config-vlan)# exit
Ruijie(config)# interface vlan 1
Ruijie(config-if-VLAN 1)# ip address 10.10.10.10 255.255.255.0
Ruijie(config-if-VLAN 1)# exit
Ruijie(config)# packet double-distribute arp interface aggregatePort 1
```

- 【检验方法】 首先通过 **show run** 命令查看镜像是否正确配置 配置成功后查看接入交换机 Switch1 和 Switch2 的 api 表项，能够正常看到接入服务器 Server1 和 Server2 的表项。

常见错误

配置命令报错时，请检查 AP 口成员数量。

6.6 监视与维护


清除各类信息

无。

查看运行情况

作用	命令
查看系统存在的所有镜像会话。	show monitor
查看具体的镜像会话。	show monitor session <i>session-id</i>

查看调试信息

 输出调试信息，会占用系统资源。使用完毕后，请立即关闭调试开关。

作用	命令
打开 SPAN 的调试开关。	debug span

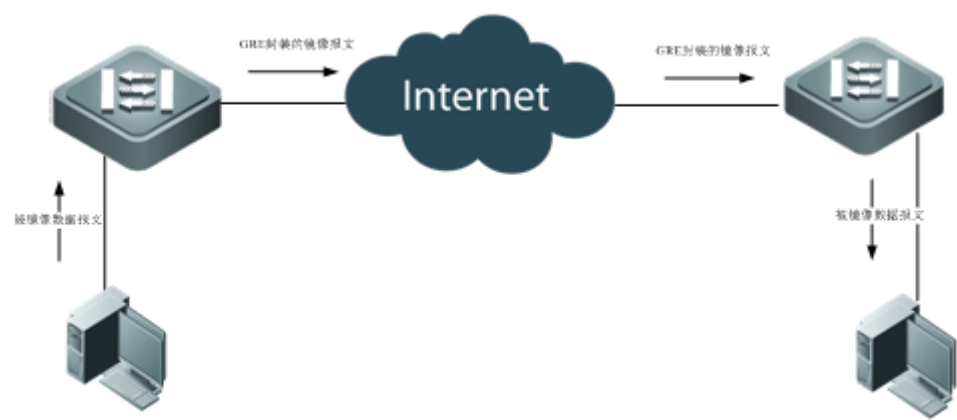
7 ERSPAN

7.1 概述

封装远程端口镜像(ERSPAN)是远程端口镜像(RSPAN)的扩展。普通的远程端口镜像，镜像数据报文只能在二层内传输，无法经过路由的网络，而封装远程端口镜像却可以将镜像报文在路由的网络间传输。

ERSPAN 实现的功能是将所有的被镜像报文通过一个 GRE 隧道封装成 IP 报文，路由到远端镜像设备的目的端口，典型应用拓扑如下所示：

图 7-1 ERSPAN 典型应用拓扑图



图中各设备的角色分为两种：

- 源交换机：封装远程镜像源端口所在的交换机，负责将源端口的报文复制一份从源交换机的输出端口输出，通过 GRE 封装成 IP 报文进行转发，传输给目的交换机。
- 目的交换机：封装远程镜像目的端口所在的交换机，将接收到的镜像报文通过镜像目的端口，进行解封装 GRE 报文后转发给监控设备。

要实现封装远程端口镜像功能，进行的 GRE 封装后的 IP 报文是必须可以在网络中正常路由到目的镜像设备的。

协议规范

- 无

7.2 典型应用

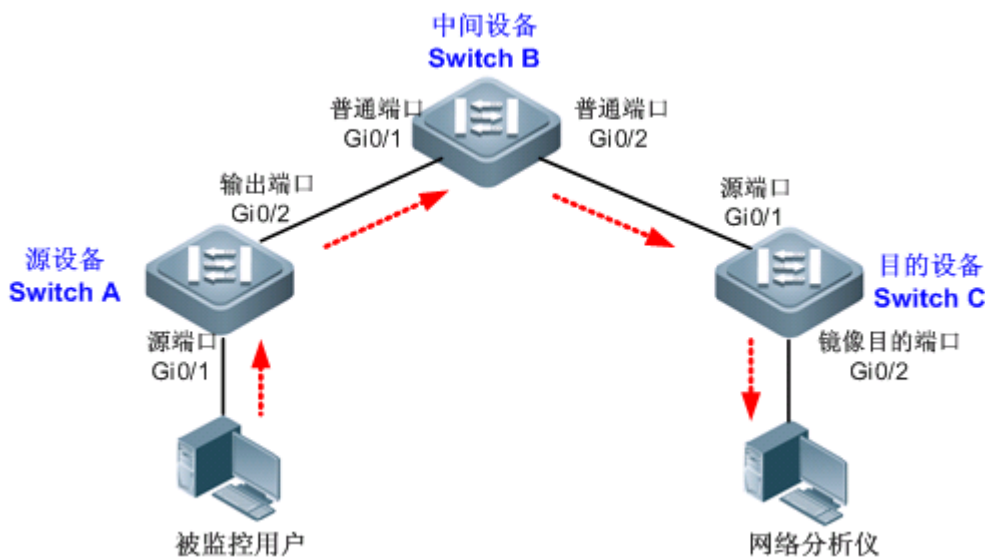
典型应用	场景描述
ERSPAN 基本应用	需要将镜像源设备的报文镜像到目的设备上上进行监控。

7.2.1 ERSPAN 基本应用

应用场景

如图所示，网络分析仪可以通过 ERSPAN 功能，实现监控连接到源设备 Switch A 上的用户。且设备之间均能正常交换数据。

图 7-2 ERSPAN 基本应用拓扑



【注释】 -

功能部署

- 在源设备上，配置直连用户的端口（本例为 Gi 0/1）为源端口，与中间设备相连的端口（本例为 Gi 0/2）为输出端口。
- 在中间设备上，与源设备、目的设备相连的端口（本例为 Gi 0/1 和 Gi 0/2）分别为两个网段的 SVI 口的成员口，并保证这两个 IP 网段可以互通。

7.3 功能详解

基本概念

ERSPAN 会话

普通的远程端口镜像，镜像数据报文只能在二层内传输，无法经过路由的网络，而 ERSPAN 镜像却可以将镜像报文在路由的网络间传输。ERSPAN 实现的功能是将所有的被镜像报文通过一个 GRE 隧道封装成 IP 报文，路由到远端镜像设备的目的端

口。ERSPAN 可以监控单个或多个端口的输入、输出、双向的报文。Switched Port、Routed Port 和 AP(聚合端口)等类型的端口都可以配置为 ERSPAN 会话的源端口。端口加入 ERSPAN 会话后并不影响交换机的正常操作。

源端口

源端口也被称为被监控口，在 ERSPAN 会话中，源端口上的数据流被监控，用于网络分析或故障排查。在单个 ERSPAN 会话中，用户可以监控输入、输出或双向数据流，且源端口的个数没有限制。源端口具有以下特性：

- 源端口可以是 switched port，routed port 或 AP(Aggregate Port)；
- 支持将源设备上的多个源端口镜像到指定的输出端口；
- 源端口与输出端口不能为同一端口；当镜像源端口为三层接口时，监控的报文包括二层报文和三层报文；
- 在双向监控多个端口的情况下，一份报文由一个端口进入，从另外一个端口输出，只要有监控到一份报文视为正确；
- 当启用 STP 的端口处于 block 状态时，该端口输入、输出的报文能够被监控到；
- 源端口和目的端口可以属于同一 VLAN，也可以属于不同 VLAN。

功能特性

功能特性	作用
ERSPAN	跨 Internet 网络端口的镜像。

7.3.1 ERSPAN

封装远程端口镜像(ERSPAN)是远程端口镜像(RSPAN)的扩展。普通的远程端口镜像，镜像数据报文只能在二层内传输，无法经过路由的网络，而封装远程端口镜像却可以将镜像报文在路由的网络间传输。

工作原理

将所有的被镜像报文通过一个 GRE 隧道封装成 IP 报文，路由到远端镜像设备的目的端口。

配置 ERSPAN 会话

配置交换机设备的 ERSPAN 功能，区分设备的 ERSPAN 交换机的属性。用户需要指定镜像会话 ID，配置成功后会进入 ERSPAN 配置模式。

配置源端口

进入 ERSPAN 配置模式后，用户需要指定源端口名字来配置镜像源端口，并通过镜像方向的可选配置项决定镜像数据流的方向。

ERSAN 会话使能

对 ERSPAN 会话使能，默认是开启 ERSPAN 镜像功能。只有处于使能状态的 ERSPAN 会话才会生效。

封装源 IP 地址

封装源 IP 地址是用来设置封装的 GRE 报文的源 IP 地址。

✎ 封装目的 IP 地址

封装目的 IP 地址是用来设置封装的 GRE 报文的目的 IP 地址，保证镜像报文可以正常的在网络中路由。

✎ 封装 ip ttl/dscp

封装 IP 报文的 TTL 和 DSCP 值。

✎ vrf vrf-name

虚拟路由的名字，不同的虚拟路由的值对于相同的目的 ip 获取到的出口可能不一样。

相关配置

系统镜像功能默认是关闭的，只有用户创建会话，并配置源镜像端口和源 IP、目的 IP 才会开启镜像功能。

✎ 配置 ERSPAN 会话

Ruijie(config)# **monitor session** *session_num* **erspan-source**

其中，

session-num：镜像会话 ID，针对不同产品支持镜像会话个数会有所不同。

✎ 配置源端口

Ruijie(config-mon-erspan-src)# **source interface** *single_interface* {[**rx** | **tx** | **both**]}

其中，

single_interface：待配置的镜像源端口。

rx：配置 **rx** 选项后，只监听源端口接收的报文。

tx：配置 **tx** 选项后，只监听源端口发送的报文。

both：配置 **both** 选项后，源端口收发的报文都会送到目的端口进行监听，即包含 **rx** 和 **tx**。如果用户不配置 **rx**、**tx** 和 **both** 三个选项中的任何一个则默认开启 **both** 选项。

✎ 配置基于流的镜像

缺省情况下，该功能关闭。用户通过 Ruijie(config-mon-erspan-src)# **source interface** *interface-id* **rx acl** *acl-name* 命令配置基于流的镜像。

✎ ERSPAN 会话使能

Ruijie (config-mon-erspan-src)# **shutdown**

关闭 ERSPAN 镜像功能。(默认)开启 ERSPAN 镜像功能，使用 **no shutdown** 命令。

✎ 封装目的 IP 地址

Ruijie(config-mon-erspan-src)# **destination ip address** *ip-address*

其中，

ip-address：封装目的 IP 地址

↘ 封装源 IP 地址

```
Ruijie(config-mon-erspan-src)# origin ip address ip-address
```

其中，

ip-address：封装源 IP 地址

↘ 封装 ip ttl

```
Ruijie(config-mon-erspan-src)# ip ttl ttl_value
```

其中，

ttl_value：配置封装 IP 的 ttl 值，ttl 值的范围为 0-255，默认值为 64

↘ 封装 ip dscp

```
Ruijie(config-mon-erspan-src)# ip dscp dscp_value
```

其中，


dscp_value：配置封装 IP 的 dscp 值，dscp 值的范围为 0-63，默认值为 0，该功能只有在镜像源端口配置了信任 dscp 时才生效。

↘ 封装 vrf *vrf-name*

```
Ruijie(config-mon-erspan-src)# vrf vrf-name
```

其中，

vrf-name：vrf 的名字

 使用过程中，用户需要特别注意以下几点：


- 确认从源交换机到目的交换机的三层路由互通性。
- 如果禁用了源端口，ERSPAN 将不起作用。
- 如果将源端口或目的端口加入 AP，源端口或目的端口将退出 ERSPAN 会话。
- 产品的差异性，并不是所有产品都支持上述命令的所有选项。

7.4 产品说明



S6000E 支持 ERSPAN 功能。

7.5 配置详解

配置项	配置建议 & 相关命令	
配置 ESPAN 基本功能	 必须配置。用于创建 ERSPAN 镜像。	
	Ruijie# configure terminal	开启全局配置模式
	Ruijie (config)# monitor session erspan_source_session_number erspan-source	配置一个 ERSPAN 会话号，并进入 ERSPAN 源镜像设备的配置模式。
	Ruijie (config-mon-erspan-src)# source interface single_interface {[rx tx both]}	关联 ERSPAN 镜像的源端口，并选择镜像的方向。
	Ruijie (config-mon-erspan-src)# source interface single_interface rx acl acl-name	配置 ERSPAN 基于流的镜像源
	Ruijie (config-mon-erspan-src)# shutdown	关闭 ERSPAN 镜像功能。
	Ruijie (config-mon-erspan-src)# destination ip address ip_address	配置 ERSPAN 流目的 IP 地址。该地址必须是目的设备上的接口地址。
	Ruijie (config-mon-erspan-src)# original ip address ip_address	配置 ERSPAN 封装源 IP 地址。
	Ruijie (config-mon-erspan-src)# ip ttl ttl_value	(可选)配置 ERSPAN 封装的 IP 头 TTL 值。
	Ruijie (config-mon-erspan-src)# ip dscp dscp_value	(可选)配置 ERSPAN 封装的 IP 头 dscp 字段值。
	Ruijie (config-mon-erspan-src)# vrf vrf_name	(可选) 配置 VRF 名字。

7.5.1 配置 ERSPAN 基本功能

配置效果

- 网络分析仪可以通过远程镜像监控用户。
- 设备之间均能正常交换数据。

注意事项

- 如果将源端口加入 AP，源端口将退出 ERSPAN 会话。
- 保证从源交换机到目的交换机的三层路由互通性

配置方法

- **ERSPAN 会话**
- 全局模式。必须配置。
- 已经配置本地镜像或 RSPAN 的会话 ID 不能作为 ERSPAN 上的会话 ID，配置完后进入 ERSPAN 模式。

▾ 源端口

- 全局模式。必须配置。
- 配置镜像源端口时可以选择配置的镜像方向，缺省是 both 方向，即同时监测报文的接收和发送行为。

▾ ERSPAN 会话使能

- 全局模式。必须配置。
- 对 ERSPAN 会话使能，默认是开启 ERSPAN 镜像功能。只有处于使能状态的 ERSPAN 会话才会生效。

▾ 封装源 IP 地址

- 全局模式。必须配置。
- 用于封装镜像报文源 IP 地址。

▾ 封装目的 IP 地址

- 全局模式。必须配置。
- 用于封装镜像报文目的 IP 地址。

▾ 封装 ip ttl/dscp

- 全局模式。可选。
- 用于封装镜像 IP 报文的 dscp 值。

▾ vrf vrf-name

- 全局模式。可选。
- vrf 的名字，vrf 必须存在。

检验方法

- 镜像配置的校验也可以通过 **show monitor** 或者 **show running** 命令查看。也可以在目的设备的镜像目的的口上进行抓包分析，通过抓取的报文查看镜像功能是否生效。

相关命令

▾ 配置 ERSPAN 会话

【命令格式】 **monitor session session_number erspan-source**

【参数说明】 **session-num**：镜像会话 ID

【命令模式】 全局模式

【使用指导】 -

配置源端口

【命令格式】 **source interface** *single_interface* {[**rx** | **tx** | **both**]}

【参数说明】 *single_interface* : 镜像会话 ID

both : 同时监控输入和输出方向的报文, 为缺省值

rx : 监控输入方向的报文

tx : 监控输出方向的报文

【命令模式】 ERSPAN 会话模式

【使用指导】 -

配置基于流的镜像

【命令格式】 Ruijie (config-mon-erspan-src)# **source interface** *interface-id* **rx acl** *acl-name*

【参数说明】 *interface-id* : 接口名字

acl-name : acl 名字

【命令模式】 全局模式

【使用指导】 -

ERSAN 会话使能

【命令格式】 Ruijie (config-mon-erspan-src)# **shutdown**

【参数说明】

【命令模式】 ERSPAN 会话模式

【使用指导】 -

封装源 IP 地址

【命令格式】 **original ip address** *ip_address*

【参数说明】 *ip_address* : 需要封装的源 IP 地址

【命令模式】 ERSPAN 会话模式

【使用指导】

封装目的 IP 地址

【命令格式】 **destination ip address** *ip_address*

【参数说明】 *ip_address* : 需要封装的目的 IP 地址

【命令模式】 ERSPAN 会话模式

【使用指导】

封装 ip ttl

- 【命令格式】 **ip ttl** *ttl_value*
- 【参数说明】 *ttl_value*：配置 ERSPAN 封装的 IP 头 TTL 值。
- 【命令模式】 ERSPAN 会话模式
- 【使用指导】 -

▾ 封装 dscp

- 【命令格式】 **ip dscp** *dscp_value*
- 【参数说明】 *dscp_value*：配置 ERSPAN 封装的 IP 头 dscp 字段值。
- 【命令模式】 ERSPAN 会话模式
- 【使用指导】 -

▾ 配置 vrf *vrf-name*

- 【命令格式】 **vrf** *vrf_name*
- 【参数说明】 *vrf_name*：VRF 名字
- 【命令模式】 ERSPAN 会话模式
- 【使用指导】 -

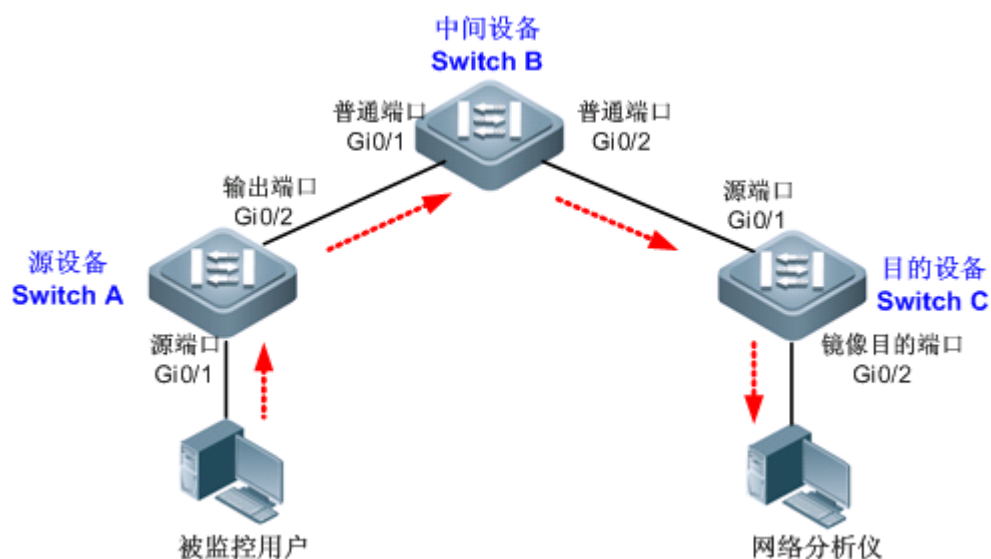
配置举例

i 以下配置举例，仅介绍与 ERSPAN 镜像相关的配置。

▾ 下面以本地镜像为例介绍

【网络环境】

图 7-3



- 【配置方法】
- 如图 1-2，在 Switch A 上，创建 ERSPAN Session 1，设置为源设备，并设置端口 Gi 0/1 为源端口。

```
SwitchA(config)#monitor session 1 erspan-source
SwitchA(config-mon-erspan-src)#source interface gigabitEthernet 0/1 both
SwitchA(config-mon-erspan-src)#origin ip address 10.1.1.2
SwitchA(config-mon-erspan-src)#destination ip address 12.1.1.2
SwitchA(config-mon-erspan-src)#vrf vrf-name
```

- 【检验方法】
- 第一步，查看设备配置信息。

```
SwitchA#show running-config
!
monitor session 1 erspan-src
  source interface GigabitEthernet 0/1 both
  origin ip address 10.1.1.2
  destination ip address 12.1.1.2
  vrf vrf-name
```

第二步，查看设备的 ERSPAN 信息

```
SwitchA#show monitor
sess-num: 1                               //ERSPAN Session
span-type: ERSPAN_SOURCE                  //ERSPAN 源设备
src-intf:                                 //ERSPAN 源端口信息
GigabitEthernet 0/1      frame-type Both  TX status: Inactive  RX status: Inactive
dest-intf:                                                         //ERPSAN 输出端口信息
GigabitEthernet 0/2
origin ip address 10.1.1.2
destination ip address 12.1.1.2
ip ttl 64
ip dscp 0
vrf vrf-name
```

常见错误

- 用户配置 ERSPAN 镜像的会话 ID 已经被配置了 RSPAN 或 LOCAL SPAN。
- 从源交换机到目的交换机的三层路由无法互通。

7.6 监视与维护


清除各类信息

无。

查看运行情况

作用	命令
查看系统存在的所有镜像会话。	show monitor
查看具体的镜像会话。	show monitor session <i>session-id</i>

查看调试信息

 输出调试信息，会占用系统资源。使用完毕后，请立即关闭调试开关。

作用	命令
打开 SPAN 的调试开关。	debug span

8 sFlow

8.1 概述

sFlow 是由 InMon、HP 和 FoundryNetworks 于 2001 年联合开发的一种网络监测技术,目前已经完成标准化,可提供完整的第二层到第四层信息,可以适应超大网络流量环境下的流量分析,让用户详细、实时地分析网络传输流的性能、趋势和存在的问题。

sFlow 具有如下优势：

- 支持在千兆或更高速的网络上精确地监控网络流量。
- 一个 sFlow Collector 能够监控成千上百个 sFlow Agent,具有良好的扩展性。
- sFlow Agent 内嵌在网络设备中,成本较低。

协议规范

- sFlow Version 5 : sFlow V5 协议。
- RFC 1014 : sFlow 使用的数据标准。

8.2 典型应用

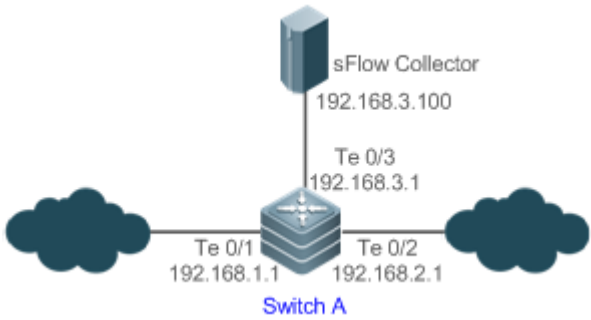
典型应用	场景描述
监控局域网流量	将设备作为 sFlow Agent ,在局域网中对接口流量进行采样 ,并将采样结果发送给 sFlow Collector 用于流量分析,以达到监控网络的目的。

8.2.1 监控局域网流量

应用场景

如图所示,启动作为 sFlow Agent 设备的交换机 SwitchA,在 Te0/1 口开启 flow 采样、counter 采样,监控 192.168.1.0 网段的流量,定时或者缓冲区满时将采样结果封装成 sFlow 报文,发送给 sFlow Collector 用于分析 sFlow Agent 监控的流量。

图 8-1



功能部署

- 在 Switch A 上配置 sFlow Agent、sFlow Collector 地址
- 在 Switch A 的 Te0/1 口开启 flow 采样、counter 采样

i 支持 sflow 的服务器软件有很多，可以在 <http://www.sflow.org/products/collectors.php> 获得，其中 sflowtrend 是免费软件。

8.3 功能详解

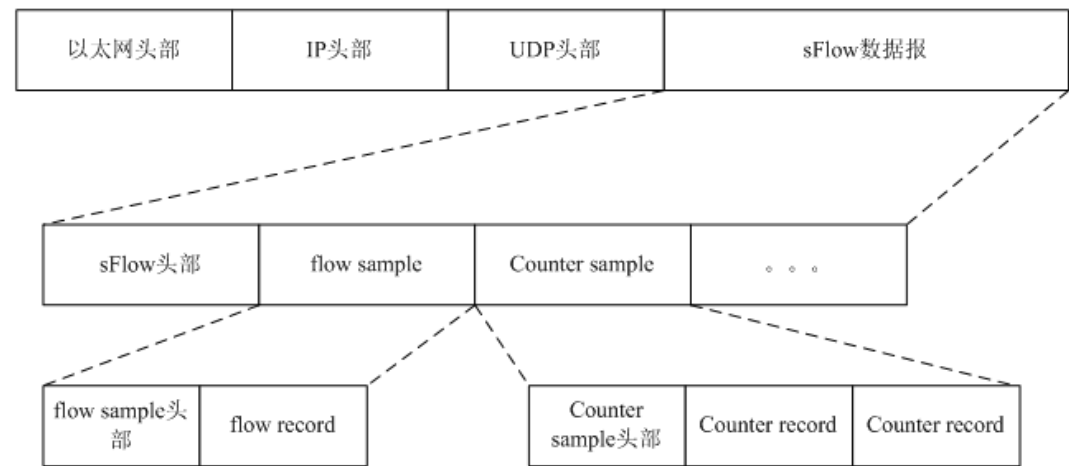
基本概念

📄 sFlow Agent

嵌入于网络设备中，通常一台网络设备可以设置成一个 sFlow Agent。sFlow Agent 可以进行 flow 采样和 counter 采样，并将采样信息封装成 sFlow 报文发送到 sFlow Collector。

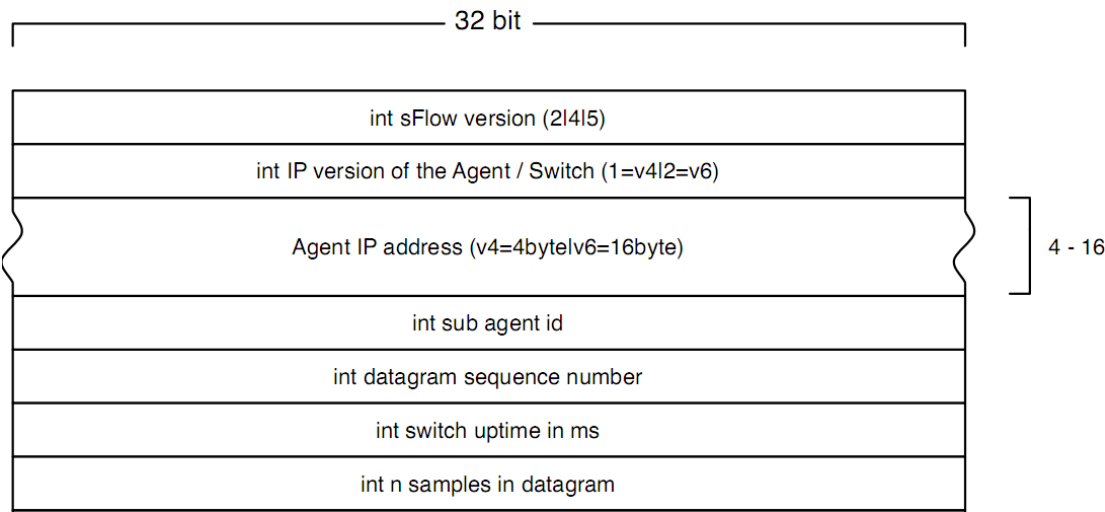
sFlow 报文采用 UDP 封装，报文格式如下图所示。

图 8-2 sFlow 报文格式



一个 sFlow 数据报可以包含一个或者多个 flow sample 和 counter sample。

图 8-3 sFlow 头部



sFlow 头部说明：

字段	说明
sFlow version	sFlow 版本号，有 2、4、5，目前锐捷只支持 v5
IP version of the agent/switch	SFlow Agent IP 地址的版本号
Agent IP address	SFlow Agent IP 地址
Sub agent id	Sub agent id
Datagram sequence number	sFlow 报文序列号
Switch uptime	交换机起机到当前经历了多少毫秒
n samples in datagram	报文中有多少个 samples，一个 sFlow 数据报可以包含一个或者多个 flow sample 和 counter sample

📌 sFlow Collector

接收 sFlow Agent 发送过来的 sFlow 报文，并进行分析。sFlow Collector 可以是 PC 或者服务器，在 PC 或者服务器上安装针对 sFlow 报文进行分析的软件即为一台 sFlow Collector。

📌 flow 采样

flow 采样是 sFlow Agent 设备在指定接口上按照特定采样率对报文进行采样分析，分析的内容包括：拷贝报文头部、提取以太网头部信息、提取路由信息等。

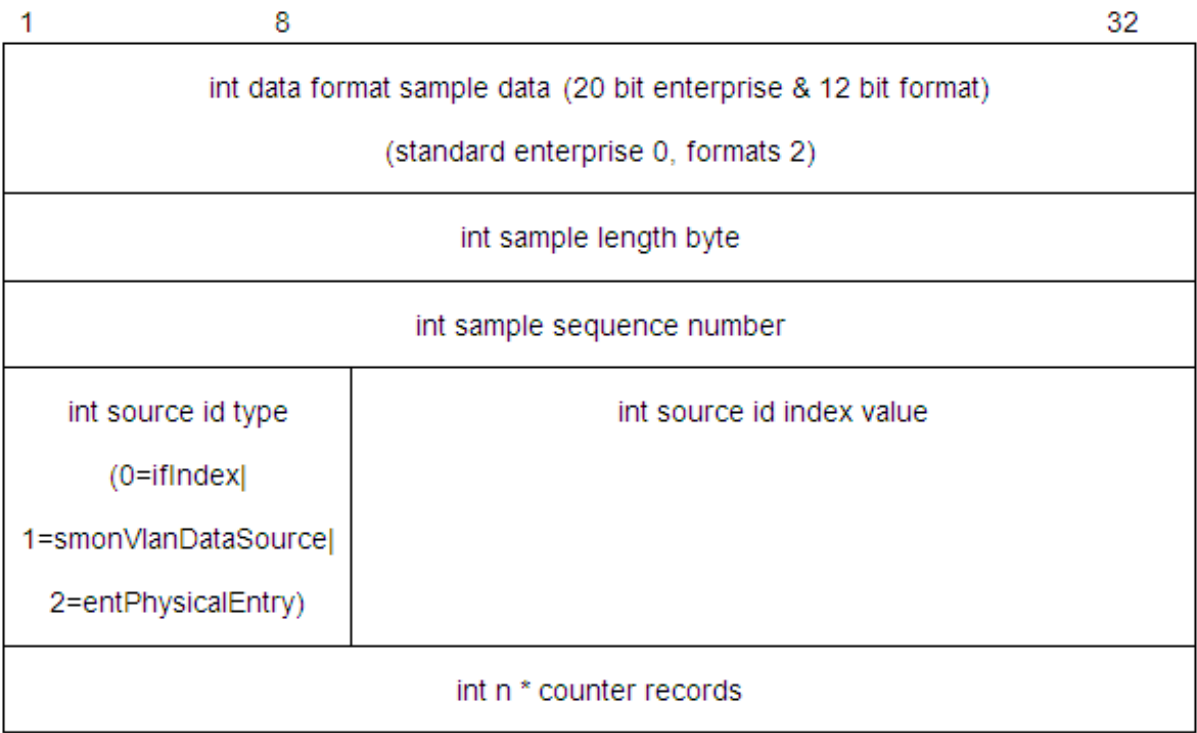
图 8-4 flow sample 头部

1	8	32
int data format sample data (20 bit enterprise & 12 bit format) (standard enterprise 0, formats 1)		
int sample length byte		
int sample sequence number		
int source id type (0=ifIndex 1=smonVlanDataSource 2=entPhysicalEntry)	int source id index value	
int sampling rate		
int sample pool (total number of packets that could have been sampled)		
int drops (packets dropped due to a lack of resources)		
int input (SNMP ifIndex of input interface, 0 if not known)		
int output (SNMP ifIndex of output interface, 0 if not known) broadcast or multicast are handled as follows: the first bit indicates multiple destinations, the lower order bits number of interfaces		
int n * flow records		

counter 采样

counter 采样是 sFlow Agent 设备周期性的获取指定接口上的统计信息、CPU 利用率。其中接口上的统计信息包括接口输入报文数、输出报文数等信息。

图 8-5 counter sample 头部



功能特性

功能特性	作用
flow 采样	对流经接口的报文进行处理，并发往 sFlow Collector 处理。
counter 采样	定时将接口的统计信息发往 sFlow Collector 处理。

8.3.1 flow 采样

对流经接口的报文进行处理，并发往 sFlow Collector 处理。

工作原理

当一个报文通过某个接口时，sFlow Agent 设备根据该接口下的采样率配置对报文进行 flow 采样，包括拷贝报文的头部、提取报文的以太网头部、IP 头部、获得报文的路由信息等。最后 sFlow Agent 模块将 flow 采样结果封装成 sFlow 报文，发送到 sFlow Collector 进行分析。

8.3.2 counter 采样

定时将接口的统计信息发往 sFlow Collector 处理。

工作原理

sFlow Agent 模块定时轮询接口，对于 counter 采样时间间隔到期的接口获得该接口的统计信息，然后将统计信息封装成 sFlow 报文，发送到 sFlow Collector 进行分析。

8.4 产品说明



S6000E 系列产品支持 SFLOW 流统计功能。



S6000E 系列产品 SFLOW 采样比统一为 (4096-65535)。

SFLOW 采样比通过设置芯片的采样阈值寄存器实现，采样阈值寄存器值等于最大采样基数(0xffff)除以 SFLOW 采样率后取整，只有相除之后余数为整数的采样率设置才是准确的，所以存在采样比设置越大可能偏差越大的情况。

S6000E 系列产品在采样率范围内，任意采样比误差均不超过 10%；

8.5 配置详解

配置项	配置建议 & 相关命令	
配置 sFlow 基本功能	⚠ 必须配置。用于建立 sFlow Agent 和 sFlow Collector 连接通信。	
	sflow agent {address interface}	配置 sFlow Agent 地址
	sflow collector collector-id destination	配置 sFlow Collector 地址
	⚠ 必须配置。用于开启 flow 采样和 counter 采样。	
	sflow counter collector	配置 counter 采样输出 sFlow Collector 的 ID
	sflow flow collector	配置 flow 采样输出 sFlow Collector 的 ID
	sflow enable	配置接口 sFlow 采样使能，同时开启 counter 采样和 flow 采样
配置 sFlow 可选参数	⚠ 可选配置。用于修改 sFlow 相关参数属性。	
	sflow collector collector-id max-datagram-size	配置输出 sFlow 报文最大长度
	sflow counter interval	配置 counter 采样时间间隔
	sflow flow max-header	配置 flow 采样拷贝报文头的最大长度
	sflow sampling-rate	配置 flow 采样的采样率

8.5.1 配置 sFlow 基本功能

配置效果

- sFlow Agent 设备同 sFlow Collector 之间可以通信。
- 根据缺省的采样率对流经接口的报文进行处理，并发往 sFlow Collector 处理。
- 根据缺省的采样间隔定时将接口的统计信息发往 sFlow Collector 处理。

注意事项

- 支持在物理口和聚合口下配置 flow 采样。
- 为使 sFlow Collector 可以对 flow 采样的结果进行分析，sFlow Agent 设备上必须配置 sFlow Collector 的 IP 地址。

配置方法

配置 sFlow Agent 地址

- 必须配置。
- 使用 **sflow agent { address | interface }**可配置 sFlow Agent 地址。
- sFlow Agent 地址必须是有效的地址。不能是组播、广播地址等。建议使用 sFlow Agent 设备的 IP 地址。

【命令格式】 **sflow agent { address { *ip-address* | ipv6 *ipv6-address* } } | { interface { *interface-name* | ipv6 *interface-name* } }**

【参数说明】 **address**：通过地址的形式配置 sFlow Agent 地址。

ip-address：sFlow Agent IPv4 地址。

ipv6 *ipv6-address*：sFlow Agent IPv6 地址。

interface：通过接口的形式配置 sFlow Agent 地址

interface-name：配置了 IPv4 地址的接口名称。

ipv6 *interface-name*：配置了 IPv6 地址的接口名称。

【缺省配置】 缺省未配置

【命令模式】 全局模式

【使用指导】 该命令用于配置填充在输出报文的 Agent ip address 字段，未配置报文将无法输出。地址只能为主机地址，当配置为非主机地址，比如组播地址、广播地址，将提示配置失败。建议配置的地址为 sFlow Agent 设备上的地址。

配置 sFlow Collector 地址

- 必须配置。
- 使用 **sflow collector** 命令可以配置 sFlow Collector 地址。
- sFlow Collector 地址必须是有效的地址。不能是组播、广播地址等。sFlow Collector 必须存在并且路由可达。

【命令格式】 **sflow collector collector-id destination { *ip-address* | ipv6 *ipv6_address* } udp-port [[vrf *vrf-name*] | [oob]]**

【参数说明】 *collector-id*：sFlow Collector id，取值范围 1-2。

ip-address：sFlow Agent IPv4 地址，缺省未配置。

ipv6 *ipv6-address* : sFlow Agent IPv6 地址，缺省未配置。

udp-port : sFlow Collector 监听端口号。

vrf *vrf-name* : VRF 实例名，缺省未配置。

oob : 采样报文从管理口输出，缺省未配置。

【命令模式】 全局模式

【使用指导】 该命令用于配置 sFlow Collector 地址，地址只能为主机地址，当配置为非主机地址，比如组播地址、广播地址，将提示配置失败。sFlow Collector 在配置的端口号上监听 sFlow 报文。
当配置了 VRF 实例时，对应的 VRF 实例必须存在。当删除了对应 VRF 实例，如果有 sFlow Collector 地址也配置了改 VRF 实例，则这个地址将被删除。

配置 flow 采样输出 sFlow Collector 的 ID

- 必须配置。
- 使用 **sflow flow collector** 命令可以启动或关闭接口上 flow 采样的输出 sFlow Collector 功能。
- 必须在接口上启用 flow 采样输出 sFlow Collector 功能，才会将接口上的 flow 采样输出到 sFlow Collector。并且 sFlow Collector 必须是存在、可达的，sFlow Agent 设备上必须已经配置了相应 sFlow Collector 的 IP 地址。

【命令格式】 **sflow flow collector** *collector-id*

【参数说明】 *collector-id* : sFlow Collector id，取值范围 1-2。

【缺省配置】 接口上 flow 采样的输出 sFlow Collector 功能关闭。

【命令模式】 接口模式

【使用指导】 该命令支持在物理口、SVI 口、聚合口和路由口下配置。
对应的 sFlow Collector 只有配置 IP 地址，sFlow 报文才能输出。

配置 counter 采样输出 sFlow Collector 的 ID

- 必须配置。
- 使用 **sflow counter collector** 命令可以启动或关闭接口上 counter 采样的输出 sFlow Collector 功能。
- 必须在接口上启用 counter 采样输出 sFlow Collector 功能，才会将接口上的 counter 采样输出到 sFlow Collector。并且 sFlow Collector 必须是存在、可达的，sFlow Agent 设备上必须已经配置了相应 sFlow Collector 的 IP 地址。

【命令格式】 **sflow counter collector** *collector-id*

【参数说明】 *collector-id* : sFlow Collector id，取值范围 1-2。

【缺省配置】 接口上 counter 采样的输出 sFlow Collector 功能关闭。

【命令模式】 接口模式

【使用指导】 该命令支持在物理口、SVI 口、聚合口和路由口下配置。
对应的 sFlow Collector 只有配置 IP 地址，sFlow 报文才能输出。

开启 counter 采样和 flow 采样

- 必须配置。

- 使用 **sflow enable** 命令可以开启接口上的 flow 采样功能以及 counter 采样功能。
- 开启 flow 采样可能影响接口的转发性能。

- 【命令格式】 **sflow enable**
- 【参数说明】 -
- 【缺省配置】 接口上 flow 采样功能关闭
- 【命令模式】 接口模式
- 【使用指导】 该命令支持在物理口、聚合口下配置。

检验方法

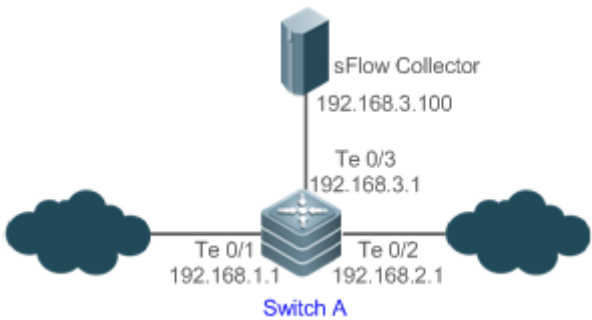
- 利用命令 **show sflow** 显示 sFlow 配置信息，查看显示信息是否与配置一致。

配置举例

配置 sFlow Agent 的 flow 采样和 counter 采样

【网络环境】

图 8-6



如图所示，启动作为 sFlow Agent 设备的交换机 SwitchA，在 Te0/1 口开启 flow 采样、counter 采样，监控 192.168.1.0 网段的流量，定时或者缓冲区满时将采样结果封装成 sFlow 报文，发送给 sFlow Collector 用于分析 sFlow Agent 监控的流量。

- 【配置方法】
 - 配置 sFlow Agent 地址为 192.168.1.1。
 - 配置 sFlow Collector 1 地址为 192.168.3.100，端口号为 6343。
 - 在接口 TenGigabitEthernet 0/1 配置 flow 采样、counter 采样输出到 sFlow Collector 1，并使能该接口的 sFlow 采样功能。

Switch A

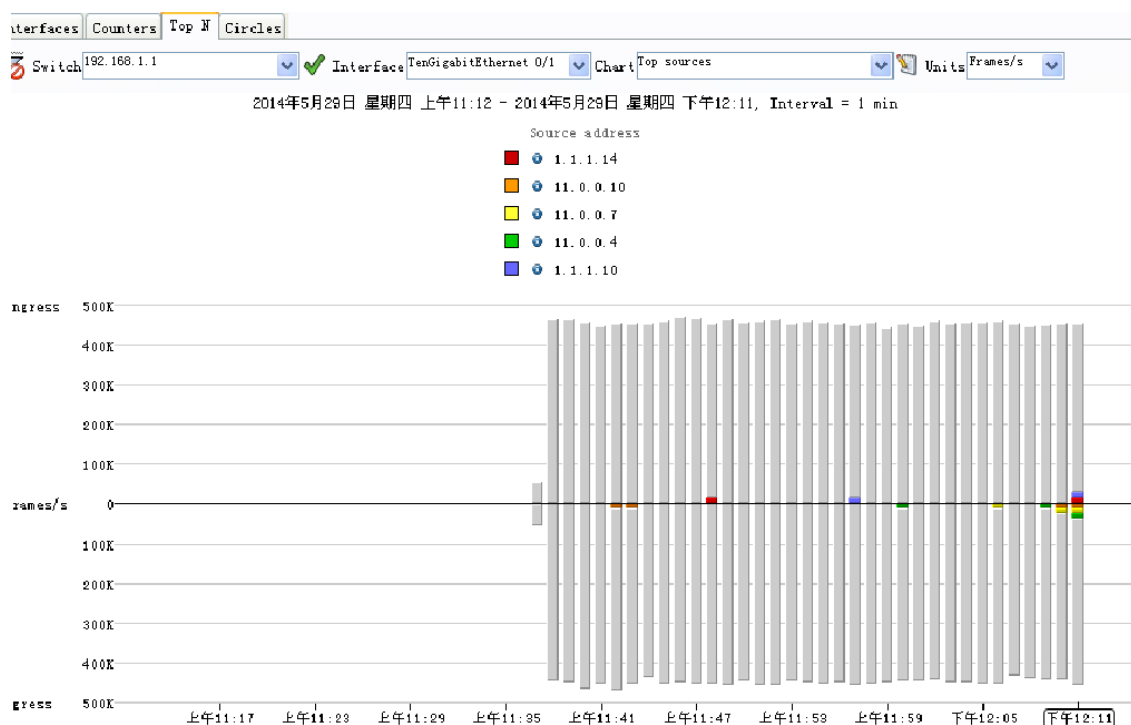
```
Ruijie# configure terminal
Ruijie(config)# sflow agent address 192.168.1.1
Ruijie(config)# sflow collector 1 destination 192.168.3.100 6343
Ruijie(config)# interface TenGigabitEthernet 0/1
Ruijie(config-if-TenGigabitEthernet 0/1)# sflow flow collector 1
Ruijie(config-if-TenGigabitEthernet 0/1)# sflow counter collector 1
Ruijie(config-if-TenGigabitEthernet 0/1)# sflow enable
Ruijie(config-if-TenGigabitEthernet 0/1)# end
```

- 【检验方法】 通过 **show sflow** 查看显示信息是否与配置一致。

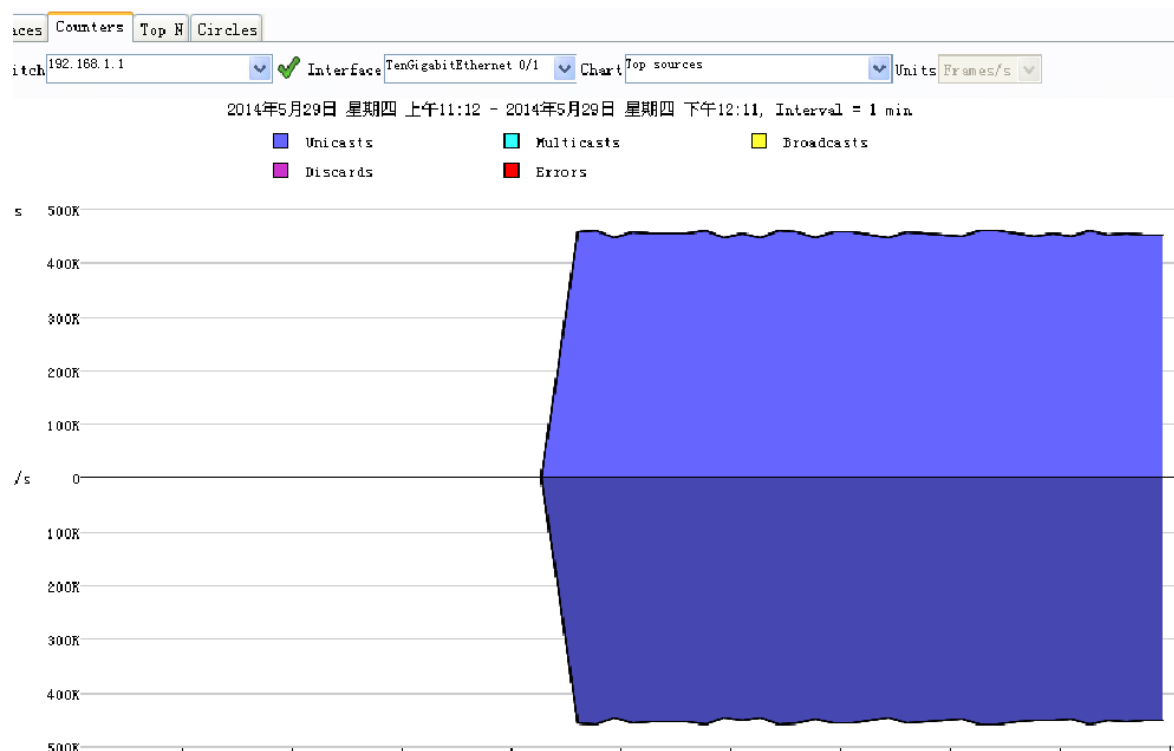
```
Ruijie# show sflow
```

```
sFlow datagram version 5
Global information:
Agent IP: 192.168.1.1
sflow counter interval:30
sflow flow max-header:64
sflow sampling-rate:8192
Collector information:
ID  IP                               Port Size VPN
1   192.168.3.100                     6343 1400
2   NULL                               0    1400
Port information
Interface          CID  FID  Enable
TenGigabitEthernet 0/1    1    1    Y
```

sflowtrend 上的显示



上图是 sflowtrend 的 TOP N 界面，用于显示 flow 采样结果，显示了流量最大前 5 个源 IP 地址，总流量为入方向 450KPPS 左右、出方向 450KPPS，与实际流量一致。



上图是 sflowtrend 的 counters 界面，用于显示 counter 采样结果，入方向 450KPPS，出方向 450KPPS,且所有报文均为单播报文。

常见错误

-

8.5.2 配置 sFlow 可选参数

配置效果

通过修改 sFlow 相关参数属性的缺省值，可调节数据采样的精确度。

注意事项

- 采样率配置太低可能影响转发性能。

配置方法

配置输出 sFlow 报文最大长度

- 可选配置。

- 使用 **sflow collector** 命令可以配置 sFlow 报文载荷的长度，不包括以太网头部、IP 头部、UDP 头部。sflow 报文中可以封装 1 个或者多个 flow 采样和 counter 采样。sflow 输出报文最大长度配置会导致处理同样数量的 flow 采样和 counter 采样输出的 sflow 报文个数可能不一样。如果配置超过 MTU，输出的 sflow 报文会被分片。

【命令格式】 **sflow collector collector-id max-datagram-size datagram-size**

【参数说明】 *collector-id*：sFlow Collector id，取值范围 1-2。

max-datagram-size datagram-size：输出 sFlow 报文最大长度，取值范围 200-9000。

【缺省配置】 缺省值 1400

【命令模式】 全局模式

【使用指导】 -

📌 配置 sFlow flow 采样的采样率

- 可选配置。
- 使用 **sflow sampling-rate** 命令可以配置全局 flow 采样的采样率。
- flow 采样的采样率配置可能影响到 sflow 采样的准确性，采样率越小，准确度越高，同时也越消耗 CPU，从而有可能影响到接口转发性能。

【命令格式】 **sflow sampling-rate rate**

【参数说明】 *rate*：sFlow flow 采样的采样率，即每 *rate* 个报文采样一个报文，取值范围为 4096-16777215。

【缺省配置】 全局的 flow 采样的采样率为 8192。

【命令模式】 全局模式

【使用指导】 该命令配置了 sFlow flow 采样的全局采样率，所有接口的 sFlow flow 采样都使用这个采样率。

📌 配置 flow 采样拷贝报文头的最大长度

- 可选配置。
- 使用 **sflow flow max-header** 命令可以配置全局 flow 采样拷贝报文的长度。
- 用户可以通过该配置修改输出到 sFlow Collector 的报文信息。例如，用户关心 IP 头部，则可以配置长度为 56 字节。封装 flow 采样时将采样报文的前 56 个字节复制到 sflow 报文中。

【命令格式】 **sflow flow max-header length**

【参数说明】 *Length*：拷贝报文头最大长度，取值范围 18-256，缺省值 64，单位字节。

【缺省配置】 全局的 flow 采样拷贝报文的长度为 64 字节。

【命令模式】 全局模式

【使用指导】 配置在进行报文内容拷贝时，从原始报文的头部开始，允许拷贝的最大字节数。拷贝的内容会记录在生成的采样样本中。

📌 配置采样时间间隔

- 可选配置。
- 使用 **sflow counter interval** 命令可以配置全局的 counter 采样时间间隔。
- 使能 counter 采样的接口每隔采样时间间隔就会将接口的统计信息发送到 sflow collector。

- 【命令格式】 **sflow counter interval seconds**
- 【参数说明】 *seconds* : 时间间隔, 取值范围 3-2147483647, 单位为秒, 缺省值 30。
- 【缺省配置】 全局的 counter 采样时间间隔为 30 秒。
- 【配置模式】 全局模式
- 【使用指导】 该命令配置了 sFlow counter 采样的全局时间间隔, 所有接口的 sFlow counter 采样都使用这个采样间隔。

检验方法

- 在 sFlow Collector 上观察是否收到内容为 flow 采样的 sFlow 报文。
- 利用命令 **show sflow** 显示 sFlow 配置信息, 查看显示信息是否与配置一致。

配置举例

配置 sflow 可选配置

- 【网络环境】 参见图 8-6
- 【配置方法】
- 在全局模式下配置采样率为 4096。
 - 在全局模式下配置拷贝报文头的前 128 个字节。
 - 在全局模式下配置采样间隔为 10。

```
Ruijie# configure terminal
Ruijie(config)# sflow sampling-rate 4096
Ruijie(config)# sflow flow max-header 128
Ruijie(config)# sflow counter interval 10
```

- 【检验方法】 使 TenGigabitEthernet 0/1 有流量经过。
- 在 sFlow Collector 1 中观察 TenGigabitEthernet 0/1 是否有流量
 - 通过 **show sflow** 查看显示信息是否与配置一致

```
Ruijie# show sflow
sFlow datagram version 5
Global information:
Agent IP: 10.10.10.10

sflow counter interval:10

sflow flow max-header:128

sflow sampling-rate:4096

Collector information:
ID   IP                      Port Size VPN
1    192.168.2.100          6343 1400
2    NULL                   0    1400

Port information
Interface          CID  FID  Enable
TenGigabitEthernet 0/1    0    1    Y
```

常见错误

-

8.6 监视与维护

清除各类信息

-

查看运行情况

作用	命令
查看 sFlow 配置。	show sflow

查看调试信息

-