

# H3C S5130S-SI[LI]&S5120V2-SI[LI]&S5110V2-SI& S5000V3-EI&S5000E-X&S3100V3-SI 系列以太网交换机

## 网络管理和监控命令参考

新华三技术有限公司  
<http://www.h3c.com>

资料版本：6W103-20190822  
产品版本：Release 612x 系列

**Copyright © 2019 新华三技术有限公司及其许可者 版权所有，保留一切权利。**

未经本公司书面许可，任何单位和个人不得擅自摘抄、复制本书内容的部分或全部，并不得以任何形式传播。

除新华三技术有限公司的商标外，本手册中出现的其它公司的商标、产品标识及商品名称，由各自权利人拥有。

# 前言

本命令参考主要介绍了网络管理和监控相关的配置命令。通过这些命令您可以对网络进行管理和监控，包括查看系统信息、对网络流量进行统计、对网络质量进行分析、对网络内所有具有时钟的设备进行时钟同步，并可以使用 **ping**、**tracert**、**debug** 等命令来检查、调试当前网络的连接情况。

前言部分包含如下内容：

- [读者对象](#)
- [本书约定](#)
- [资料意见反馈](#)

## 读者对象

本手册主要适用于如下工程师：

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

## 本书约定

### 1. 命令行格式约定

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






### 2. 图形界面格式约定

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

格 式	意 义
	的[文件夹]菜单项。

### 3. 各类标志

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

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

### 4. 图标约定

本书使用的图标及其含义如下：

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



该图标及其相关描述文字代表防火墙插卡、负载均衡插卡、NetStream插卡、SSL VPN插卡、IPS插卡、ACG插卡等安全插卡。

## 5. 示例约定

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

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

## 资料意见反馈

如果您在使用过程中发现产品资料的任何问题，可以通过以下方式反馈：

E-mail: [info@h3c.com](mailto:info@h3c.com)

感谢您的反馈，让我们做得更好！

# 目 录

1 系统维护与调试.....	1-1
1.1 系统维护与调试命令.....	1-1
1.1.1 debugging.....	1-1
1.1.2 display debugging.....	1-2
1.1.3 ping .....	1-2
1.1.4 ping ipv6.....	1-5
1.1.5 tracert .....	1-7
1.1.6 tracert ipv6 .....	1-8

# 1 系统维护与调试

## 1.1 系统维护与调试命令

### 1.1.1 debugging

**debugging** 命令用来打开指定模块的调试开关。

**undo debugging** 命令用来关闭指定模块的调试开关。

#### 【命令】

```
debugging module-name [ option ]  
undo debugging { all | module-name [ option ] }
```

#### 【缺省情况】

所有模块的调试开关均处于关闭状态。

#### 【视图】

用户视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**module-name**: 模块名称，比如 arp、device 等。可以使用 **debugging ?** 命令查询设备当前支持的模块名。

**option**: 模块的调试选项。对于不同的模块，调试选项的数量和内容都不相同。可以使用 **debugging module-name ?** 命令查询设备当前支持的指定模块的调试选项。

**all**: 所有模块的调试开关。仅当需要关闭所有调试开关时可使用本参数。

#### 【使用指导】

调试信息的输出会影响系统的运行效率，所以建议在进行网络故障诊断时根据需要打开某个功能模块的调试开关，不要同时打开多个功能模块的调试开关。

执行本命令后设备会将生成的调试信息发送到设备的信息中心模块，通过设置信息中心的参数，最终决定调试信息的输出规则（即是否允许输出以及输出方向）。有关调试信息输出规则的详细介绍请参见“网络管理和监控配置指导”中的“信息中心”。

#### 【举例】

# 打开设备管理模块的调试开关。

```
<Sysname> debugging dev
```

#### 【相关命令】

- **display debugging**

### 1.1.2 display debugging

**display debugging** 命令用来显示系统中已经打开的调试开关。

#### 【命令】

```
display debugging [ module-name ]
```

#### 【视图】

任意视图

#### 【缺省用户角色】

```
network-admin  
network-operator
```

#### 【参数】

*module-name*: 显示指定模块调试开关的设置情况。*module-name* 表示模块名，具体取值可通过执行 **display debugging ?** 命令来获取。如果不指定本参数，则显示所有打开的调试开关。

#### 【举例】

```
# 显示所有打开的调试开关。  
<Sysname> display debugging  
DEV debugging switch is on
```

#### 【相关命令】

- **debugging**

### 1.1.3 ping

**ping** 命令用来检查指定 IP 地址是否可达，并输出相应的统计信息。

#### 【命令】

```
ping [ ip ] [ -a source-ip | -c count | -f | -h ttl | -i interface-type  
interface-number | -m interval | -n | -p pad | -q | -r | -s packet-size | -t  
timeout | -tos tos | -v ] * host
```

#### 【视图】

任意视图

#### 【缺省用户角色】

```
network-admin
```

#### 【参数】

**ip**: 支持 IPv4 协议以及区分有特定字母的目的主机名。不指定该参数时，也表示支持 IPv4 协议。如果 **ping** 的目的主机名为 **i**、**ip**、**ipv**、**ipv6**、**l**、**ls**、**lsp** 时，需要先指定该关键字再指定主机名，如：**ping ip ip**。

**-a source-ip**: 指定 ICMP 回显请求（ECHO-REQUEST）报文的源 IP 地址。该地址必须是设备上已配置的 IP 地址。不指定该参数时，ICMP 回显请求报文的源 IP 地址是该报文出接口的主 IP 地址。



**-c count:** 指定 ICMP 回显请求报文的发送次数，取值范围为 1~4294967295，缺省值为 5。

**-f:** 将长度大于出接口 MTU 的报文直接丢弃，即不允许对发送的 ICMP 回显请求报文进行分片。

**-h ttl:** 指定 ICMP 回显请求报文中的 TTL 值，取值范围为 1~255，缺省值为 255。

**-i interface-type interface-number:** 指定发送 ICMP 回显请求报文的接口的类型和编号。不指定该参数时，将根据目的 IP 查找路由表或者转发表来确定发送 ICMP 回显请求报文的接口。

**-m interval:** 指定发送 ICMP 回显请求报文的时间间隔，取值范围为 1~65535，单位为毫秒，缺省值为 200。

**-n:** 对 *host* 参数不进行域名解析。不指定该参数时，如果 *host* 参数表示的是目的端的主机名，则设备会对 *host* 进行域名解析。

**-p pad:** 指定 ICMP 回显请求报文的“PAD”字段的填充值，为 1~8 位的 16 进制数，取值范围为 0~FFFFFFFF。如果指定的参数不够 8 位，则会在首部补 0，使填充值达到 8 位。比如将 *pad* 设置为 0x2f，则会重复使用 0x0000002f 去填充报文，以使发送报文的总长度达到设备要求值。填充值从 0x01 开始，逐渐递增，直到 0xff，然后又从 0x01 开始循环，形如 0x010203……feff01……，直至发送报文的总长度达到设备要求值。

**-q:** 只显示统计信息。不指定该参数时，系统将显示包括统计信息在内的全部信息。

**-r:** 记录路由信息。不指定该参数时，系统不记录路由。

**-s packet-size:** 指定发送的 ICMP 回显请求报文的长度（不包括 IP 和 ICMP 报文头），取值范围为 20~8100，单位为字节，缺省值为 56。

**-t timeout:** 指定 ICMP 回显应答（ECHO-REPLY）报文的超时时间，发送 ICMP 回显请求报文 *timeout* 时长后还没有收到 ICMP 回显应答报文，源端则认为 ICMP 回显应答报文超时。取值范围为 0~65535，单位为毫秒，缺省值为 2000。

**-tos tos:** 指定 ICMP 回显请求报文中的 ToS 域的值，取值范围为 0~255，缺省值为 0。

**-v:** 显示接收到的非回显应答的 ICMP 报文。不指定该参数时，系统不显示非回显应答的 ICMP 报文。

**host:** 目的端的 IP 地址或主机名。其中，主机名为 1~253 个字符的字符串，不区分大小写，字符串仅可包含字母、数字、“-”、“\_”或“.”。

## 【使用指导】

如果要使用目的端的主机名执行 ping 操作，事先必须在设备上配置 DNS（Domain Name System，域名系统）功能，否则会 ping 失败。

在执行命令过程中，键入<Ctrl+C>可终止 ping 操作。

## 【举例】

# 检查 IP 地址为 1.1.2.2 的设备是否可达。

```
<Sysname> ping 1.1.2.2
Ping 1.1.2.2 (1.1.2.2): 56 data bytes, press CTRL_C to break
56 bytes from 1.1.2.2: icmp_seq=0 ttl=254 time=2.137 ms
56 bytes from 1.1.2.2: icmp_seq=1 ttl=254 time=2.051 ms
56 bytes from 1.1.2.2: icmp_seq=2 ttl=254 time=1.996 ms
56 bytes from 1.1.2.2: icmp_seq=3 ttl=254 time=1.963 ms
56 bytes from 1.1.2.2: icmp_seq=4 ttl=254 time=1.991 ms

--- Ping statistics for 1.1.2.2 ---
```

```
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 1.963/2.028/2.137/0.062 ms
# 检查 IP 地址为 1.1.2.2 的设备是否可达，只显示检查结果。
<Sysname> ping -q 1.1.2.2
Ping 1.1.2.2 (1.1.2.2): 56 data bytes, press CTRL_C to break

--- Ping statistics for 1.1.2.2 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 1.962/2.196/2.665/0.244 ms
# 检查 IP 地址为 1.1.2.2 的设备是否可达，并显示路由信息。
<Sysname> ping -r 1.1.2.2
Ping 1.1.2.2 (1.1.2.2): 56 data bytes, press CTRL_C to break
56 bytes from 1.1.2.2: icmp_seq=0 ttl=254 time=4.685 ms
RR:      1.1.2.1
         1.1.2.2
         1.1.1.2
         1.1.1.1
56 bytes from 1.1.2.2: icmp_seq=1 ttl=254 time=4.834 ms (same route)
56 bytes from 1.1.2.2: icmp_seq=2 ttl=254 time=4.770 ms (same route)
56 bytes from 1.1.2.2: icmp_seq=3 ttl=254 time=4.812 ms (same route)
56 bytes from 1.1.2.2: icmp_seq=4 ttl=254 time=4.704 ms (same route)

--- Ping statistics for 1.1.2.2 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 4.685/4.761/4.834/0.058 ms
以上显示信息表明本设备与 1.1.2.2 之间路由可达，具体路由为 1.1.1.1 <-> {1.1.1.2; 1.1.2.1} <-> 1.1.2.2。
```

表1-1 ping 命令显示信息描述表

字段	描述
Ping 1.1.2.2 (1.1.2.2): 56 data bytes, press CTRL_C to break	检查IP地址为1.1.2.2的设备是否可达。每个ICMP回显请求报文中的数据为56字节，按组合键Ctrl+C可以终止ping操作
56 bytes from 1.1.2.2: icmp_seq=0 ttl=254 time=4.685 ms	收到IP地址为1.1.2.2的设备回复的ICMP响应报文，若超时仍没有收到ICMP响应报文，则不输出信息 <ul style="list-style-type: none"><li>bytes 表示 ICMP 响应报文中的数据字节数</li><li>icmp_seq 表示报文序号，用来判断报文是否有分组丢失、失序或重复</li><li>ttl 表示 ICMP 响应报文中的 TTL 值</li><li>time 表示响应时间</li></ul>
RR:	ICMP回显请求报文经过的路由器，采用倒序显示，距离目的端越近的路由器越先显示
--- Ping statistics for 1.1.2.2 ---	Ping操作中收发数据的统计结果
5 packet(s) transmitted	发送的ICMP回显请求报文数
5 packet(s) received	收到的ICMP响应报文数

字段	描述
0.0% packet loss	未响应请求报文占发送的总请求报文的百分比
round-trip min/avg/max/std-dev = 4.685/4.761/4.834/0.058 ms	响应时间的最小值、平均值、最大值和标准方差，单位为毫秒

### 1.1.4 ping ipv6

**ping ipv6** 命令用来检查指定 IPv6 地址是否可达，并输出相应的统计信息。

#### 【命令】

```
ping ipv6 [ -a source-ipv6 | -c count | -i interface-type interface-number |
-m interval | -q | -s packet-size | -t timeout | -tc traffic-class | -v ] *
host
```

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**-a source-ipv6**: 指定 ICMPv6 回显请求报文中的源 IPv6 地址。该地址必须是设备上已配置的合法 IPv6 地址。不指定该参数时，ICMPv6 回显请求报文的源 IPv6 地址是该报文出接口的地址（地址选择原则遵循 RFC 3484）。

**-c count**: 指定发送的 ICMPv6 回显请求报文的数目，取值范围为 1~4294967295，缺省值为 5。

**-i interface-type interface-number**: 指定出接口的接口类型与接口编号。对端是组播地址或者是链路本地地址则必须指定此参数。不指定该参数时，将根据目的 IP 查找路由表或者转发表来确定发送 ICMPv6 回显请求报文的接口。

**-m interval**: 指定发送 ICMPv6 回显请求报文的时间间隔，取值范围为 1~65535，单位为毫秒，缺省值为 1000。

**-q**: 只显示统计信息。不指定该参数时，系统将显示包括统计信息在内的全部信息。

**-s packet-size**: 指定发送的 ICMPv6 回显请求报文的长度（不包括 IPv6 和 ICMPv6 报文头），取值范围为 20~8100，单位为字节，缺省值为 56。

**-t timeout**: 指定 ICMPv6 回显应答报文的超时时间，取值范围为 0~65535，单位为毫秒，缺省值为 2000。

**-tc traffic-class**: IPv6 ICMP 报文中的 Traffic Class 域的值。取值范围为 0~255，缺省值为 0。

**-v**: 显示 ICMPv6 回显应答报文的详细信息。不指定该参数时，显示 ICMPv6 回显应答报文的简要信息。详细信息比简要信息多 **dst** 和 **idx** 字段，**dst** 表示回显应答报文的目的地址，**idx** 表示回显应答报文的入接口索引。

**host**: 目的端的 IPv6 地址或主机名。其中，主机名为 1~253 个字符的字符串，不区分大小写，字符串仅可包含字母、数字、“-”、“\_”或“.”。

【使用指导】

如果要使用目的端的主机名执行 **ping ipv6** 操作，事先必须在设备上配置 DNS 功能，否则 IPv6 ping 操作将会失败。

在执行命令过程中，键入<Ctrl+C>可终止 **ping ipv6** 操作。

【举例】

# 检查 IPv6 地址为 2001::2 的设备是否可达。

```
<Sysname> ping ipv6 2001::2
Ping6(56 data bytes) 2001::1 --> 2001::2, press CTRL_C to break
56 bytes from 2001::2, icmp_seq=0 hlim=64 time=62.000 ms
56 bytes from 2001::2, icmp_seq=1 hlim=64 time=23.000 ms
56 bytes from 2001::2, icmp_seq=2 hlim=64 time=20.000 ms
56 bytes from 2001::2, icmp_seq=3 hlim=64 time=4.000 ms
56 bytes from 2001::2, icmp_seq=4 hlim=64 time=16.000 ms
```

```
--- Ping6 statistics for 2001::2 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 4.000/25.000/62.000/20.000 ms
```

# 检查 IPv6 地址为 2001::2 的设备是否可达，只显示统计信息。

```
<Sysname> ping ipv6 -q 2001::2
Ping6(56 data bytes) 2001::1 --> 2001::2, press CTRL_C to break
```

```
--- Ping6 statistics for 2001::2 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 4.000/25.000/62.000/20.000 ms
```

# 检查 IPv6 地址为 2001::2 的设备是否可达，显示详细 ping 信息。

```
<Sysname> ping ipv6 -v 2001::2
Ping6(56 data bytes) 2001::1 --> 2001::2, press CTRL_C to break
56 bytes from 2001::2, icmp_seq=0 hlim=64 dst=2001::1 idx=3 time=62.000 ms
56 bytes from 2001::2, icmp_seq=1 hlim=64 dst=2001::1 idx=3 time=23.000 ms
56 bytes from 2001::2, icmp_seq=2 hlim=64 dst=2001::1 idx=3 time=20.000 ms
56 bytes from 2001::2, icmp_seq=3 hlim=64 dst=2001::1 idx=3 time=4.000 ms
56 bytes from 2001::2, icmp_seq=4 hlim=64 dst=2001::1 idx=3 time=16.000 ms
```

```
--- Ping6 statistics for 2001::2 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 4.000/25.000/62.000/20.000 ms
```

以上信息表明，目的端可达，源端发出的 ICMPv6 回显请求报文均能得到回应，报文往返时间的最小值、平均值、最大值和标准方差分别为 4ms、25ms、62ms 和 20ms。

表1-2 ping ipv6 命令显示信息描述表

字段	描述
Ping6 (56 data bytes) 2001::1 --> 2001::2, press CTRL_C to break	从源地址2001::1给目的地址2001::2发送一个ICMPv6回显请求报文，每个ICMPv6回显请求报文中的数据为56字节，按组合键Ctrl+C可以终止IPv6 ping操作

字段	描述
56 bytes from 2001::2, icmp_seq=1 hlim=64 dst=2001::1 idx=3 time=62.000 ms	收到IPv6地址为2001::2的设备回复的ICMPv6响应报文，其中： <ul style="list-style-type: none"> <li>• 数据字节数为 56</li> <li>• 报文序号为 1</li> <li>• hop limit 值为 64</li> <li>• 目的地址为 2001::1（使用-v 参数时才显示该字段）</li> <li>• 报文入接口的索引为 3（使用-v 参数时才显示该字段）</li> <li>• 响应时间是 62ms</li> </ul>
--- Ping6 statistics for 2001::2 ---	IPv6 ping操作中收发数据的统计结果
5 packet(s) transmitted	发送的ICMPv6回显请求报文数
5 packet(s) received	收到的ICMPv6响应报文数
0.0% packet loss	未响应请求报文占发送的总请求报文的百分比
round-trip min/avg/max/ std-dev =4.000/25.000/62.000/20.000 ms	响应时间的最小值、平均值、最大值和标准方差，单位为毫秒

### 1.1.5 tracer

**tracer** 命令用来查看 IPv4 报文从源端传到目的端所经过的路径。

#### 【命令】

```
tracer [ -a source-ip | -f first-ttl | -m max-ttl | -p port | -q packet-number  
| -t tos | -w timeout ] * host
```

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**-a source-ip**: 指定 tracer 报文的源 IP 地址。该地址必须是设备上已配置的合法 IP 地址。不指定该参数时，tracer 报文的源 IP 地址是该报文出接口的主 IP 地址。

**-f first-ttl**: 指定一个初始 TTL，即第一个报文所允许的最大跳数。取值范围为 1~255，且小于或等于最大 TTL，缺省值为 1。

**-m max-ttl**: 指定一个最大 TTL，即一个报文所允许的最大跳数。取值范围为 1~255，且大于或等于初始 TTL，缺省值为 30。

**-p port**: 指定目的端的 UDP 端口号，取值范围为 1~65535，缺省值为 33434。用户一般不需要更改此选项。

**-q packet-number**: 指定每次发送的探测报文个数，取值范围为 1~65535，缺省值为 3。

**-t tos**: Tracer 报文中 ToS 域的值。取值范围为 0~255，缺省值为 0。

**-w timeout**: 指定探测报文的响应报文的超时时间，取值范围是 1~65535，单位为毫秒，缺省值为 5000。

host: 目的端的 IP 地址或主机名。其中，主机名为 1~253 个字符的字符串，不区分大小写，字符串仅可包含字母、数字、“-”、“\_”或“.”。

【使用指导】

当用户使用 ping 命令测试发现网络出现故障后，可以用 tracert 命令分析出现故障的网络节点。tracert 命令的输出信息包括到达目的端所经过的所有三层设备的 IP 地址，如果某设备不能回应 ICMP 错误消息（可能因为路由不可达或者没有开启 ICMP 错误报文处理功能），则输出“\* \* \*”。在执行命令过程中，键入<Ctrl+C>可终止此次 tracert 操作。

【举例】

```
# 查看报文从源端到目的端（IP 地址为 1.1.2.2）所经过的路径。
<Sysname> tracert 1.1.2.2
tracert to 1.1.2.2 (1.1.2.2), 30 hops at most, 40 bytes each packet, press CTRL_C to break
 1  1.1.1.2 (1.1.1.2) 673 ms 425 ms 30 ms
 2  1.1.2.2 (1.1.2.2) [AS 100] 580 ms 470 ms 80 ms
```

表1-3 tracert 命令显示信息描述表

字段	描述
tracert to 1.1.2.2 (1.1.2.2)	查看IP报文从当前设备传到地址为1.1.2.2的设备所经过的路径
hops at most	探测报文的最大跳数，可使用-m参数配置
bytes each packet	探测报文字节数
press CTRL_C to break	在执行命令过程中，键入<Ctrl+C>可终止此次tracert操作
2 1.1.2.2 (1.1.2.2) [AS 100] 580 ms 470 ms 80 ms	TTL值为2的探测报文的探测结果，内容包括：第二跳的域名（如果没有配置域名则显示IP地址）、IP地址、所属AS号（如果不存在则不显示）、三份探测报文的往返时间 每次发送探测报文的份数可以使用-q参数配置
1 1.1.1.2 (1.1.1.2) 673 ms 425 ms 30 ms	TTL值为1的探测报文的探测结果，内容包括：第一跳的域名（如果没有配置域名则显示IP地址）、IP地址、三份探测报文的往返时间 每次发送探测报文的份数可以使用-q参数配置

1.1.6 tracert ipv6

tracert ipv6 命令用来查看 IPv6 报文从源端传到目的端所经过的路径。

【命令】

```
tracert ipv6 [ -f first-hop | -m max-hops | -p port | -q packet-number | -t traffic-class | -w timeout ] * host
```

【视图】

任意视图

【缺省用户角色】

network-admin

【参数】

-f first-hop: 指定一个初始 hoplimit，即第一个报文所允许的跳数。取值范围为 1~255，且小于或等于 max-hops，缺省值为 1。

-m max-hops: 指定一个最大 hoplimit，即一个报文所允许的最大跳数。取值范围为 1~255，且大于或等于 first-hop，缺省值为 30。

-p port: 指定目的端的 UDP 端口号，取值范围为 1~65535，缺省值为 33434。用户一般不需要更改此选项。

-q packet-number: 指定每次发送的探测报文个数，取值范围为 1~65535，缺省值为 3。

-t traffic-class: IPv6 tracert 报文中的 Traffic Class 域的值。取值范围为 0~255，缺省值为 0。

-w timeout: 指定探测报文的响应报文的超时时间，取值范围为 1~65535，单位为毫秒，缺省值为 5000。

host: 目的端的 IPv6 地址或主机名。其中，主机名为 1~253 个字符的字符串，不区分大小写，字符串仅可包含字母、数字、“-”、“\_”或“.”。

【使用指导】

当用户使用 ping ipv6 命令测试发现网络出现故障后，可以用 tracert ipv6 命令来帮助查找出现故障的网络节点。

本命令的输出信息包括到达目的端所经过的所有三层设备的 IPv6 地址，如果某设备不能回应 ICMP 错误消息（可能因为路由不可达或者没有开启 ICMP 错误报文处理功能），则输出 “\* \* \*”。

在执行命令过程中，键入<Ctrl+C>可终止此次 tracert ipv6 操作。

【举例】

# 查看报文从源端到目的端（IPv6 地址为 2001:3::2）所经过的路径。

```
<Sysname> tracert ipv6 2001:3::2
tracert to 2001:3::2(2001:3::2), 30 hops at most, 60 byte packets , press CTRL_C to break
 1  2001:1::2  0.661 ms  0.618 ms  0.579 ms
 2  2001:2::2 [AS 100]  0.861 ms  0.718 ms  0.679 ms
 3  2001:3::2 [AS 200]  0.822 ms  0.731 ms  0.708 ms
```

表1-4 tracert ipv6 命令显示信息描述表

字段	描述
tracert to 2001:3::2	查看IPv6报文从当前设备发送到地址为2001:3::2的设备所经过的路径
hops at most	探测报文的最大跳数，可使用-m参数配置
byte packets	探测报文字节数
2 2001:2::2 [AS 100] 0.861 ms 0.718 ms 0.679 ms	hoplimit值为2的探测报文的探测结果，内容包括：第二跳的IPv6地址、所属AS号（如果不存在则不显示）、三份探测报文的往返时间（每次发送探测报文的份数可以使用-q参数配置）

# 目 录

1 NQA .....	1-1
1.1 NQA客户端配置命令 .....	1-1
1.1.1 advantage-factor .....	1-1
1.1.2 codec-type .....	1-1
1.1.3 community read.....	1-2
1.1.4 data-fill .....	1-3
1.1.5 data-size.....	1-4
1.1.6 description .....	1-5
1.1.7 destination host.....	1-6
1.1.8 destination ip .....	1-6
1.1.9 destination ipv6 .....	1-7
1.1.10 destination port.....	1-8
1.1.11 display nqa history .....	1-9
1.1.12 display nqa reaction counters .....	1-10
1.1.13 display nqa result .....	1-12
1.1.14 display nqa statistics.....	1-19
1.1.15 expect data.....	1-27
1.1.16 expect ip .....	1-29
1.1.17 expect ipv6 .....	1-29
1.1.18 expect status.....	1-30
1.1.19 filename.....	1-31
1.1.20 frequency .....	1-31
1.1.21 history-record enable .....	1-32
1.1.22 history-record keep-time .....	1-33
1.1.23 history-record number .....	1-34
1.1.24 init-ttl .....	1-34
1.1.25 key .....	1-35
1.1.26 lsr-path .....	1-36
1.1.27 max-failure .....	1-36
1.1.28 mode.....	1-37
1.1.29 next-hop ip .....	1-38
1.1.30 next-hop ipv6 .....	1-38
1.1.31 no-fragment enable .....	1-39



1.1.32 nqa .....	1-40
1.1.33 nqa agent enable .....	1-40
1.1.34 nqa schedule .....	1-41
1.1.35 nqa template .....	1-42
1.1.36 operation (FTP test type view) .....	1-43
1.1.37 operation (HTTP/HTTPS test type view) .....	1-44
1.1.38 out interface .....	1-45
1.1.39 password .....	1-46
1.1.40 probe count .....	1-47
1.1.41 probe packet-interval .....	1-48
1.1.42 probe packet-number .....	1-48
1.1.43 probe packet-timeout .....	1-49
1.1.44 probe timeout .....	1-50
1.1.45 raw-request .....	1-51
1.1.46 reaction checked-element { jitter-ds   jitter-sd } .....	1-51
1.1.47 reaction checked-element { owd-ds   owd-sd } .....	1-53
1.1.48 reaction checked-element icpif .....	1-54
1.1.49 reaction checked-element mos .....	1-55
1.1.50 reaction checked-element packet-loss .....	1-56
1.1.51 reaction checked-element probe-duration .....	1-57
1.1.52 reaction checked-element probe-fail (for trap) .....	1-58
1.1.53 reaction checked-element probe-fail (for trigger) .....	1-60
1.1.54 reaction checked-element rtt .....	1-61
1.1.55 reaction trap .....	1-62
1.1.56 reaction trigger per-probe .....	1-63
1.1.57 reaction trigger probe-fail .....	1-64
1.1.58 reaction trigger probe-pass .....	1-65
1.1.59 resolve-target .....	1-66
1.1.60 resolve-type .....	1-66
1.1.61 route-option bypass-route .....	1-67
1.1.62 source interface (ICMP-echo/UDP-tracert test type view) .....	1-68
1.1.63 source ip .....	1-69
1.1.64 source ipv6 .....	1-70
1.1.65 source port .....	1-71
1.1.66 ssl-client-policy .....	1-71
1.1.67 statistics hold-time .....	1-72

1.1.68 statistics interval .....	1-73
1.1.69 statistics max-group.....	1-73
1.1.70 target-only .....	1-74
1.1.71 tos .....	1-75
1.1.72 ttl.....	1-75
1.1.73 type .....	1-76
1.1.74 url.....	1-77
1.1.75 username.....	1-78
1.1.76 version .....	1-79
1.2 NQA服务器端命令 .....	1-80
1.2.1 display nqa server.....	1-80
1.2.2 nqa server enable.....	1-81
1.2.3 nqa server tcp-connect .....	1-81
1.2.4 nqa server udp-echo .....	1-82

# 1 NQA

## 1.1 NQA客户端配置命令

### 1.1.1 advantage-factor

**advantage-factor** 命令用来配置用于计算 MOS 值和 ICPIF 值的补偿因子。

**undo advantage-factor** 命令用来恢复缺省情况。

#### 【命令】

**advantage-factor** *factor*

**undo advantage-factor**

#### 【缺省情况】

补偿因子取值为 0。

#### 【视图】

Voice 测试类型视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*factor*: 用于计算 MOS 值和 ICPIF 值的补偿因子，取值范围为 0~20。

#### 【使用指导】

用户对语音质量的评价具有一定的主观性，不同用户对语音质量的容忍程度不同，因此，衡量语音质量时，需要考虑用户的主观因素。对语音质量容忍程度较强的用户，可以通过 **advantage-factor** 命令配置补偿因子，在计算 ICPIF 值时将减去该补偿因子，修正 ICPIF 和 MOS 值，以便在比较语音质量时综合考虑客观和主观因素。

#### 【举例】

# 在 Voice 测试类型下配置计算 MOS 值和 ICPIF 值的补偿因子为 10。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type voice
[Sysname-nqa-admin-test-voice] advantage-factor 10
```

### 1.1.2 codec-type

**codec-type** 命令用来配置语音测试的编码格式。

**undo codec-type** 命令用来恢复缺省情况。

#### 【命令】

**codec-type** { *g711a* | *g711u* | *g729a* }

**undo codec-type**

### 【缺省情况】

语音编码格式为 G.711 A 律。

### 【视图】

Voice 测试类型视图

### 【缺省用户角色】

network-admin

### 【参数】

**g711a**: G.711 A 律语音编码格式。

**g711u**: G.711  $\mu$  律语音编码格式。

**g729a**: G.729 A 律语音编码格式。

### 【举例】

# 在 Voice 测试类型下配置语音测试的编码格式为 G.729 A 律。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type voice
[Sysname-nqa-admin-test-voice] codec-type g729a
```

## 1.1.3 community read

**community read** 命令用来配置用于 SNMP 测试使用的团体名。

**undo community read** 命令用来恢复缺省情况。

### 【命令】

```
community read { cipher | simple } community-name
undo community read
```

### 【缺省情况】

SNMP 测试使用的团体名为 public。

### 【视图】

SNMP 测试类型视图

### 【缺省用户角色】

network-admin

### 【参数】

**cipher**: 以密文方式配置团体名。

**simple**: 以明文方式配置团体名，该密码将以密文形式存储。

*community-name*: 测试使用的团体名，区分大小写。当以明文形式配置时，团体名为 1~32 个字符的字符串；当以密文形式配置时，团体名为 33~73 个字符的字符串。

### 【使用指导】

当被测试的 SNMP Agent 的版本为 SNMPv1 或者 SNMPv2c 时，若 SNMP Agent 配置了团体名，则 SNMP 测试组必须配置本命令，且配置的团体名必须与 SNMP Agent 上已配置的只读或读写权

限团体名保持一致,才能通过 SNMP Agent 的认证,并进行探测。当 SNMP Agent 的版本为 SNMPv3 时,由于 SNMPv3 没有团体名概念,不需要配置团体名,直接进行探测。有关团体名及 SNMP 认证的具体介绍请参见“网络管理和监控”中的“SNMP”。

#### 【举例】

# 配置 SNMP 探测报文使用的团体名称为 readaccess。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type snmp
[Sysname-nqa-admin-test-snmp] community read simple readaccess
```

### 1.1.4 data-fill

**data-fill** 命令用来配置探测报文的填充字符串。

**undo data-fill** 命令用来恢复缺省情况。

#### 【命令】

```
data-fill string
undo data-fill
```

#### 【缺省情况】

探测报文的填充内容为十六进制 00010203040506070809。

#### 【视图】

ICMP-echo/UDP-echo 测试类型视图  
Path-jitter/UDP-jitter/Voice 测试类型视图  
ICMP/TCP/UDP 类型的 NQA 模板视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*string*: 探测报文的填充内容,为 1~200 个字符的字符串,区分大小写。

#### 【使用指导】

如果探测报文的数据段长度比配置的填充数据长度小,系统在报文封装时以报文的数据段长度为界截取该字符串的前一部分;

如果探测报文的数据段长度比配置的填充数据长度大,系统在报文封装时用该字符串进行循环填充,直到填满。

例如,配置填充数据为“abcd”,当探测报文数据段长度为 3 字节时,则取“abc”作为填充数据;当探测报文大小为 6 字节时,则使用“adcdab”作为填充数据。

对于不同的测试类型,填充字符串的情况有所不同:

- 在 ICMP-echo 测试中,配置的字符串用来填充 ICMP Echo 消息的数据字段。
- 在 UDP-echo 测试中,由于 UDP 报文数据字段的前 5 个字节具有特定用途,所以只用所配置的字符串填充报文中剩余的字节。

- 在 UDP-jitter 测试中，UDP 报文数据字段的前 68 个字节具有特定用途，所以只用所配置的字符串填充报文中剩余的字节。
- 在 Voice 测试中，UDP 报文数据字段的前 16 个字节具有特定用途，所以只用所配置的字符串填充报文中剩余的字节。
- 在 Path-jitter 测试中，由于 ICMP 探测阶段 ICMP 报文数据字段的前 4 个字节具有特定用途，所以只用所配置的字符串填充 ICMP 报文中剩余的字节。

【举例】

# 在 ICMP-echo 测试类型下配置探测报文的填充字符串为 abcd。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] data-fill abcd
```

# 在 TCP 类型的 NQA 模板视图下配置探测报文的填充字符串为 abcd。

```
<Sysname> system-view
[Sysname] nqa template tcp tcptplt
[Sysname-nqatplt-tcp-tcptplt] data-fill abcd
```

1.1.5 data-size

**data-size** 命令用来配置探测报文中的填充内容的大小。

**undo data-size** 命令用来恢复缺省情况。

【命令】

```
data-size size
undo data-size
```

【缺省情况】

缺省情况如 [表 1-1](#) 所示。

表1-1 探测报文中的填充内容大小的缺省值

测试类型	编码类型	缺省值（字节）
ICMP-echo	-	100
UDP-echo	-	100
UDP-jitter	-	100
UDP-tracert	-	100
Path-jitter	-	100
Voice	G.711 A律	172
Voice	G.711 μ律	172
Voice	G.729 A律	32

【视图】

ICMP-echo/UDP-echo 测试类型视图

UDP-tracert 测试类型视图

Path-jitter/UDP-jitter/Voice 测试类型视图

ICMP/UDP 类型的 NQA 模板视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**size**: 探测报文中的填充内容的大小，单位为字节，ICMP-echo、UDP-echo 和 UDP-tracert 测试类型取值范围为 20~65507，UDP-jitter 和 Path-jitter 测试类型取值范围为 68~65507，Voice 测试类型取值范围为 16~65507。

#### 【使用指导】

对于 ICMP-echo 和 Path-jitter 测试，探测报文中填充内容的大小为 ICMP Echo 消息中数据字段的长度。

对于 UDP-echo、UDP-jitter、UDP-tracert 和 Voice 测试，探测报文中填充内容的大小为 UDP 报文中数据字段的长度。

#### 【举例】

# 在 ICMP-echo 测试类型下配置探测报文中的填充内容的大小为 80 字节。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] data-size 80
```

# 在 ICMP 类型的 NQA 模板视图下配置 ICMP-echo 探测报文中的填充内容的大小为 80 字节。

```
<Sysname> system-view
[Sysname] nqa template icmp icmptplt
[Sysname-nqatplt-icmp-icmptplt] data-size 80
```

### 1.1.6 description

**description** 命令用来配置测试组的描述信息，通常用于描述一个测试组的测试类型或测试目的。

**undo description** 命令用来恢复缺省情况。

#### 【命令】

```
description text
undo description
```

#### 【缺省情况】

未配置描述信息。

#### 【视图】

任意测试类型视图

任意类型的 NQA 模板视图

#### 【缺省用户角色】

network-admin

### 【参数】

**text**: 测试组的描述，为 1~200 个字符的字符串，区分大小写。

### 【举例】

# 在 ICMP-echo 测试类型下配置测试组的描述信息为 icmp-probe。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] description icmp-probe
```

# 在 ICMP 类型的 NQA 模板视图下配置描述信息为 icmp-probe。

```
<Sysname> system-view
[Sysname] nqa template icmp icmptplt
[Sysname-nqatplt-icmp-icmptplt] description icmp-probe
```

## 1.1.7 destination host

**destination host** 命令用来配置探测报文的目的主机名。

**undo destination host** 命令用来恢复缺省情况。

### 【命令】

**destination host** *host-name*

**undo destination host**

### 【缺省情况】

未配置探测报文的目的主机名。

### 【视图】

UDP-tracert 测试类型视图

### 【缺省用户角色】

network-admin

### 【参数】

**host-name**: 探测报文的目的主机名，为 1~254 个字符的字符串，区分大小写，字符串中可以包含字母、数字、“-”、“\_”和“.”，不能出现连续的“.”。如果目的主机名由“.”分隔成多个段，则每段不能超过 63 个字符。

### 【举例】

# 在 UDP-Tracert 测试类型下配置探测报文的目的主机名为 www.test.com。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type udp-tracert
[Sysname-nqa-admin-test-udp-tracert] destination host www.test.com
```

## 1.1.8 destination ip

**destination ip** 命令用来配置探测报文的目的 IPv4 地址。

**undo destination ip** 命令用来恢复缺省情况。



#### 【命令】

```
destination ip ip-address  
undo destination ip
```

#### 【缺省情况】

未配置探测报文的目的 IPv4 地址。

#### 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图  
DHCP/DLSw/DNS/SNMP 测试类型视图  
UDP-tracert 测试类型视图  
ICMP-jitter/Path-jitter/UDP-jitter/Voice 测试类型视图  
DNS/ICMP/RADIUS/SSL/TCP/TCP Half Open/UDP 类型的 NQA 模板视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*ip-address*：探测报文的目的 IPv4 地址。

#### 【举例】

```
# 在 ICMP-echo 测试类型下配置探测报文的目的 IPv4 地址为 10.1.1.1。  
<Sysname> system-view  
[Sysname] nqa entry admin test  
[Sysname-nqa-admin-test] type icmp-echo  
[Sysname-nqa-admin-test-icmp-echo] destination ip 10.1.1.1  
# 在 ICMP 类型的 NQA 模板视图下配置探测报文的目的 IPv4 地址为 10.1.1.1。  
<Sysname> system-view  
[Sysname] nqa template icmp icmptplt  
[Sysname-nqatplt-icmp-icmptplt] destination ip 10.1.1.1
```

### 1.1.9 destination ipv6

**destination ipv6** 命令用来配置探测报文的目的 IPv6 地址。

**undo destination ipv6** 命令用来恢复缺省情况。

#### 【命令】

```
destination ipv6 ipv6-address  
undo destination ipv6
```

#### 【缺省情况】

未配置探测报文的目的 IPv6 地址。

#### 【视图】

ICMP-echo 测试类型视图  
DNS/ICMP/RADIUS/SSL/TCP/TCP Half Open/UDP 类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

*ipv6-address*: 探测报文的目的 IPv6 地址，不支持 IPv6 链路本地地址。

### 【举例】

# 在 ICMP-echo 测试类型下配置探测报文的目的 IPv6 地址为 1::1。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] destination ipv6 1::1
```

# 在 ICMP 类型的 NQA 模板视图下配置探测报文的目的 IPv6 地址为 1::1。

```
<Sysname> system-view
[Sysname] nqa template icmp icmptplt
[Sysname-nqatplt-icmp-icmptplt] destination ipv6 1::1
```

## 1.1.10 destination port

**destination port** 命令用来配置测试操作的目的端口号。

**undo destination port** 命令用来恢复缺省情况。

### 【命令】

**destination port** *port-number*

**undo destination port**

### 【缺省情况】

对于 UDP-tracert 测试，目的端口号缺省为 33434；对于 SNMP 测试，目的端口号缺省为 161；对于其他类型测试，未配置测试操作的目的端口号。

对于各类型的 NQA 模板，各种操作类型的端口号缺省为 DNS（53）、RADIUS（1812）；对于其他模板类型，未配置测试操作的目的端口号。

### 【视图】

TCP/UDP-echo 测试类型视图

UDP-tracert 测试类型视图

UDP-jitter/Voice 测试类型视图

DNS/RADIUS/SSL/TCP/UDP 类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

*port-number*: 测试操作的目的端口号，取值范围为 1~65535。

### 【举例】

# 在 UDP-echo 测试类型视图下配置测试操作的目的端口号为 9000。

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

```
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type udp-echo
[Sysname-nqa-admin-test-udp-echo] destination port 9000
# 在 TCP 类型的 NQA 模板视图下配置测试操作的目的端口号为 9000。
<Sysname> system-view
[Sysname] nqa template tcp tcptplt
[Sysname-nqatplt-tcp-tcptplt] destination port 9000
```

### 1.1.11 display nqa history

**display nqa history** 命令用来显示 NQA 测试组的历史记录。

#### 【命令】

```
display nqa history [ admin-name operation-tag ]
```

#### 【视图】

任意视图

#### 【缺省用户角色】

```
network-admin
network-operator
```

#### 【参数】

*admin-name*: 创建 NQA 测试组的管理员名称, 为 1~32 个字符的字符串, 字符串中不能包括“-”, 不区分大小写。

*operation-tag*: 测试操作的标签, 为 1~32 个字符的字符串, 字符串中不能包括“-”, 不区分大小写。

#### 【使用指导】

**display nqa history** 命令的显示信息无法反映 ICMP-jitter、Path-jitter、UDP-jitter 和 Voice 测试的结果。如果了解 ICMP-jitter、Path-jitter、UDP-jitter 和 Voice 测试的结果, 建议通过 **display nqa result** 命令查看最近一次 NQA 测试的结果, 或通过 **display nqa statistics** 命令查看 NQA 测试的统计信息。

不指定任何参数, 则表示显示所有测试组的历史记录。

#### 【举例】

# 显示管理员名称为 administrator、测试类型标签为 tracert 的 UDP-tracert 测试项的历史记录。

```
<Sysname> display nqa history administrator tracert
NQA entry (admin administrator, tag tracert) history records:
Index      TTL  Response  Hop IP          Status          Time
1          2    328       4.1.1.1        Succeeded       2013-09-09 14:46:06.2
1          2    328       4.1.1.1        Succeeded       2013-09-09 14:46:05.2
1          2    328       4.1.1.1        Succeeded       2013-09-09 14:46:04.2
1          1    328       3.1.1.2        Succeeded       2013-09-09 14:46:03.2
1          1    328       3.1.1.1        Succeeded       2013-09-09 14:46:02.2
1          1    328       3.1.1.1        Succeeded       2013-09-09 14:46:01.2
```

# 查看管理员名称为 administrator、测试操作标签为 test 的 NQA 测试组的历史记录。

```
<Sysname> display nqa history administrator test
```

NQA entry (admin administrator, tag test) history records:

Index	Response	Status	Time
10	329	Succeeded	2011-04-29 20:54:26.5
9	344	Succeeded	2011-04-29 20:54:26.2
8	328	Succeeded	2011-04-29 20:54:25.8
7	328	Succeeded	2011-04-29 20:54:25.5
6	328	Succeeded	2011-04-29 20:54:25.1
5	328	Succeeded	2011-04-29 20:54:24.8
4	328	Succeeded	2011-04-29 20:54:24.5
3	328	Succeeded	2011-04-29 20:54:24.1
2	328	Succeeded	2011-04-29 20:54:23.8
1	328	Succeeded	2011-04-29 20:54:23.4

表1-2 display nqa history 命令显示信息描述表

字段	描述
Index	历史记录编号，一次UDP-tracert测试中的所有记录此编号一致
TTL	本次探测的TTL值
Response	测试成功时，为探测报文的往返时延；如果测试超时，则为超时时间；不能完成测试时，则为0。单位为毫秒
Hop IP	回复应答的节点IP地址
Status	测试结果的状态值，具体如下： <ul style="list-style-type: none"> <li>• Succeeded: 测试成功，接收到响应报文</li> <li>• Unknown error: 未知错误</li> <li>• Internal error: 内部错误</li> <li>• Timeout: 请求超时</li> </ul>
Time	测试完成时间

### 1.1.12 display nqa reaction counters

**display nqa reaction counters** 命令用来显示阈值告警组的当前监测结果。

#### 【命令】

**display nqa reaction counters** [ *admin-name* *operation-tag* [ *item-number* ] ]

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

*admin-name*: 创建 NQA 测试组的管理员名称，为 1~32 个字符的字符串，字符串中不能包括“-”，不区分大小写。

*operation-tag*: 测试操作的标签，为 1~32 个字符的字符串，字符串中不能包括“-”，不区分大小写。

*item-number*: 显示指定阈值告警组的当前监测结果。如果不指定该参数，将显示所有阈值告警组的当前监测结果。*item-number* 为阈值告警组的编号，取值范围为 1~10。

【使用指导】

如果 NQA 阈值告警组的阈值类型为平均值，或监测对象为 Voice 测试的 ICPIF 或 MOS 值，则显示的监测结果为无效值。

测试结束后，不会清除监测结果，即测试组启动后，监测结果会不断累加。

不指定任何参数，则表示显示所有测试组中所有阈值告警组的当前监测结果。

【举例】

# 显示 NQA 测试组管理员名称为 admin、测试操作标签为 test 的 ICMP-echo 测试组的所有阈值告警组的当前监测结果。

```
<Sysname> display nqa reaction counters admin test
NQA entry (admin admin, tag test) reaction counters:
  Index  Checked Element  Threshold Type  Checked Num  Over-threshold Num
  1      probe-duration  accumulate      12           4
  2      probe-duration  average         -           -
  3      probe-duration  consecutive     160         56
  4      probe-fail      accumulate      12           0
  5      probe-fail      consecutive     162          2
```

表1-3 display nqa reaction counters 命令显示信息描述

字段	描述
Index	阈值告警组的编号
Checked Element	监测的对象（不同的NQA测试类型支持的监测对象不同，具体请参见 <a href="#">表1-4</a> 、 <a href="#">表1-5</a> ）
Threshold Type	阈值类型
Checked Num	已监测的样本个数
Over-threshold Num	超出阈值的样本个数

表1-4 display nqa reaction counters 命令显示字段取值描述  
（DHCP/DLSw/DNS/FTP/HTTP/ICMP-echo/SNMP/TCP/UDP-echo 测试类型）

监测对象	阈值类型	监测的样本范围	Checked Num 取值	Over-threshold Num 取值
probe-duration	accumulate	启动NQA测试组后进行的探测	启动NQA测试组后已完成的探测次数	启动NQA测试组后累计的探测持续时间不在阈值范围内的探测次数
	average	-	-	-
	consecutive	启动NQA测试组后进行的探测	启动NQA测试组后已完成的探测次数	启动NQA测试组后连续的探测持续时间不在阈值范围内的探测次数

监测对象	阈值类型	监测的样本范围	Checked Num 取值	Over-threshold Num 取值
probe-fail	accumulate	启动NQA测试组后进行的探测	启动NQA测试组后已完成的探测次数	启动NQA测试组后累计的探测失败次数
	consecutive	启动NQA测试组后进行的探测	启动NQA测试组后已完成的探测次数	启动NQA测试组后连续的探测失败次数

表1-5 display nqa reaction counters 命令显示字段取值描述（ICMP-jitter/UDP-jitter/Voice 测试类型）

监测对象	阈值类型	监测的样本范围	Checked Num 取值	Over-threshold Num 取值
RTT	accumulate	启动NQA测试组后发送的报文	启动NQA测试组后已发送的报文个数	启动NQA测试组后累计的往返时间不在阈值范围内的报文个数
	average	-	-	-
jitter-DS/jitter-SD	accumulate	启动NQA测试组后发送的报文	启动NQA测试组后已发送的报文个数	启动NQA测试组后累计的单向抖动时间不在阈值范围内的报文个数
	average	-	-	-
OWD-DS/OWD-SD	-	启动NQA测试组后发送的报文	启动NQA测试组后已发送的报文个数	启动NQA测试组后单向时延不在阈值范围内的报文个数
packet-loss	accumulate	启动NQA测试组后发送的报文	启动NQA测试组后已发送的报文个数	启动NQA测试组后累计的丢包数
ICPIF/MOS（仅Voice测试支持）	-	-	-	-

### 1.1.13 display nqa result

**display nqa result** 命令用来显示最近一次 NQA 测试的结果。

#### 【命令】

**display nqa result** [ *admin-name operation-tag* ]

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

*admin-name*: 创建 NQA 测试组的管理员名称, 为 1~32 个字符的字符串, 字符串中不能包括“-”, 不区分大小写。

**operation-tag**: 为测试操作的标签，为 1~32 个字符的字符串，字符串中不能包括“-”，不区分大小写。

### 【使用指导】

不指定任何参数，则表示显示所有测试组的最近一次测试的结果。

### 【举例】

# 显示 NQA 测试组管理员名称为 **admin**、测试操作标签为 **test** 的 TCP 测试的最近一次测试结果。

```
<Sysname> display nqa result admin test
NQA entry (admin admin, tag test) test results:
  Send operation times: 1          Receive response times: 1
  Min/Max/Average round trip time: 35/35/35
  Square-Sum of round trip time: 1225
  Last succeeded probe time: 2011-05-29 10:50:33.2
Extended results:
  Packet loss ratio: 0%
  Failures due to timeout: 0
  Failures due to disconnect: 0
  Failures due to no connection: 0
  Failures due to internal error: 0
  Failures due to other errors: 0
```

# 显示 NQA 测试组管理员名称为 **admin**、测试操作标签为 **test** 的 ICMP-jitter 测试的最近一次测试结果。

```
<Sysname> display nqa result admin test
NQA entry (admin admin, tag test) test results:
  Send operation times: 10         Receive response times: 10
  Min/Max/Average round trip time: 1/2/1
  Square-Sum of round trip time: 13
  Last packet received time: 2015-03-09 17:40:29.8
Extended results:
  Packet loss ratio: 0%
  Failures due to timeout: 0
  Failures due to internal error: 0
  Failures due to other errors: 0
  Packets out of sequence: 0
  Packets arrived late: 0
ICMP-jitter results:
RTT number: 10
  Min positive SD: 0              Min positive DS: 0
  Max positive SD: 0              Max positive DS: 0
  Positive SD number: 0           Positive DS number: 0
  Positive SD sum: 0              Positive DS sum: 0
  Positive SD average: 0          Positive DS average: 0
  Positive SD square-sum: 0       Positive DS square-sum: 0
  Min negative SD: 1              Min negative DS: 2
  Max negative SD: 1              Max negative DS: 2
  Negative SD number: 1           Negative DS number: 1
  Negative SD sum: 1              Negative DS sum: 2
```

```

Negative SD average: 1
Negative SD square-sum: 1
One way results:
Max SD delay: 1
Min SD delay: 1
Number of SD delay: 1
Sum of SD delay: 1
Square-Sum of SD delay: 1
Lost packets for unknown reason: 0
Negative DS average: 2
Negative DS square-sum: 4
Max DS delay: 2
Min DS delay: 2
Number of DS delay: 1
Sum of DS delay: 2
Square-Sum of DS delay: 4
# 显示 NQA 测试组管理员名称为 admin、测试操作标签为 test 的 UDP-jitter 测试的最近一次测试结果。

```

```

<Sysname> display nqa result admin test
NQA entry (admin admin, tag test) test results:
Send operation times: 10          Receive response times: 10
Min/Max/Average round trip time: 15/46/26
Square-Sum of round trip time: 8103
Last packet received time: 2011-05-29 10:56:38.7
Extended results:
Packet loss ratio: 0%
Failures due to timeout: 0
Failures due to internal error: 0
Failures due to other errors: 0
Packets out of sequence: 0
Packets arrived late: 0
UDP-jitter results:
RTT number: 10
Min positive SD: 8                Min positive DS: 8
Max positive SD: 18               Max positive DS: 8
Positive SD number: 5             Positive DS number: 2
Positive SD sum: 75               Positive DS sum: 32
Positive SD average: 15           Positive DS average: 16
Positive SD square-sum: 1189      Positive DS square-sum: 640
Min negative SD: 8                Min negative DS: 1
Max negative SD: 24               Max negative DS: 30
Negative SD number: 4             Negative DS number: 7
Negative SD sum: 56               Negative DS sum: 99
Negative SD average: 14           Negative DS average: 14
Negative SD square-sum: 946       Negative DS square-sum: 1495
One way results:
Max SD delay: 22                  Max DS delay: 23
Min SD delay: 7                   Min DS delay: 7
Number of SD delay: 10            Number of DS delay: 10
Sum of SD delay: 125              Sum of DS delay: 132
Square-Sum of SD delay: 1805      Square-Sum of DS delay: 1988
SD lost packets: 0                DS lost packets: 0
Lost packets for unknown reason: 0

```

```

# 显示 NQA 测试组管理员名称为 admin、测试操作标签为 test 的 Voice 测试的最近一次测试结果。
<Sysname> display nqa result admin test

```



```

NQA entry (admin admin, tag test) test results:
    Send operation times: 1000          Receive response times: 0
    Min/Max/Average round trip time: 0/0/0
    Square-Sum of round trip time: 0
    Last packet received time: 0-00-00 00:00:00.0
Extended results:
    Packet loss ratio: 100%
    Failures due to timeout: 1000
    Failures due to internal error: 0
    Failures due to other errors: 0
    Packets out of sequence: 0
    Packets arrived late: 0
Voice results:
RTT number: 0
    Min positive SD: 0                Min positive DS: 0
    Max positive SD: 0                Max positive DS: 0
    Positive SD number: 0             Positive DS number: 0
    Positive SD sum: 0                Positive DS sum: 0
    Positive SD average: 0            Positive DS average: 0
    Positive SD square-sum: 0         Positive DS square-sum: 0
    Min negative SD: 0                Min negative DS: 0
    Max negative SD: 0                Max negative DS: 0
    Negative SD number: 0             Negative DS number: 0
    Negative SD sum: 0                Negative DS sum: 0
    Negative SD average: 0            Negative DS average: 0
    Negative SD square-sum: 0         Negative DS square-sum: 0
One way results:
    Max SD delay: 0                  Max DS delay: 0
    Min SD delay: 0                  Min DS delay: 0
    Number of SD delay: 0             Number of DS delay: 0
    Sum of SD delay: 0                Sum of DS delay: 0
    Square-Sum of SD delay: 0         Square-Sum of DS delay: 0
    SD lost packets: 0                DS lost packets: 0
    Lost packets for unknown reason: 1000
Voice scores:
    MOS value: 0.99                  ICPIF value: 87

```

# 显示 NQA 测试组管理员名称为 **admin**、测试操作标签为 **test** 的 Path-jitter 测试的最近一次测试结果。

```

<Sysname> display nqa result admin test
NQA entry (admin admin, tag test) test results:
Hop IP 192.168.40.210
Basic Results:
    Send operation times: 10
    Receive response times: 10
    Min/Max/Average round trip time: 1/1/1
    Square-Sum of round trip time: 10
Extended Results:
    Packet loss ratio: 0%

```

```

    Failures due to timeout: 0
    Failures due to internal error: 0
    Failures due to other errors: 0
    Packets out of sequence: 0
    Packets arrived late: 0
Path-Jitter Results:
    Jitter number: 9
        Min/Max/Average jitter: 0/0/0
    Positive jitter number: 0
        Min/Max/Average positive jitter: 0/0/0
        Sum/Square-Sum positive jitter: 0/0
    Negative jitter number: 0
        Min/Max/Average negative jitter: 0/0/0
        Sum/Square-Sum negative jitter: 0/0
Hop IP 192.168.50.209
    Basic Results:
        Send operation times: 10
        Receive response times: 10
        Min/Max/Average round trip time: 1/1/1
        Square-Sum of round trip time: 10
    Extended Results:
        Packet loss ratio: 0%
        Failures due to timeout: 0
        Failures due to internal error: 0
        Failures due to other errors: 0
        Packets out of sequence: 0
        Packets arrived late: 0
    Path-Jitter Results:
        Jitter number: 9
            Min/Max/Average jitter: 0/0/0
        Positive jitter number: 0
            Min/Max/Average positive jitter: 0/0/0
            Sum/Square-Sum positive jitter: 0/0
        Negative jitter number: 0
            Min/Max/Average negative jitter: 0/0/0
            Sum/Square-Sum negative jitter: 0/0
# 显示 NQA 测试组管理员名称为 admin、测试操作标签为 test 的 UDP-tracert 测试的最近一次测试结果。
<Sysname> display nqa result admin test
NQA entry (admin admin, tag test) test results:
    Send operation times: 6                Receive response times: 6
    Min/Max/Average round trip time: 35/35/35
    Square-Sum of round trip time: 1225
    Last succeeded probe time: 2013-09-09 14:23:24.5
    Extended results:
        Packet loss ratio: 0%
        Failures due to timeout: 0
        Failures due to internal error: 0

```

```

Failures due to other errors: 0
UDP-tracert results:
  TTL      Hop IP              Time
  1         3.1.1.1            2013-09-09 14:23:24.5
  2         4.1.1.1            2013-09-09 14:23:24.5

```

表1-6 display nqa result 命令显示信息描述

字段	描述
Data collecting in progress	测试正在进行中
Path jitter result is not available	测试没有生成结果
Send operation times	发送的探测报文数
Receive response times	收到的响应报文数
Min/Max/Average round trip time	最小/最大/平均往返时间，单位为毫秒
Square-Sum of round trip time	往返时间平方和
Last succeeded probe time	一次测试中最后一次成功探测的完成时间，如果一次测试中的探测均失败，则该时间显示为全0，UDP-jitter、Path-jitter和Voice测试中无此信息
Last packet received time	一次探测中最后一次成功收到正确响应报文的时间，如果一次探测中没有收到过正确的响应报文，则该时间显示为全0，只在UDP-jitter和Voice测试中存在此信息
Packet loss ratio	平均丢包率
Failures due to timeout	测试过程中超时的次数
Failures due to disconnect	对方强制断开连接的次数
Failures due to no connection	和对方建立连接失败的次数
Failures due to internal error	因内部错误失败的次数
Failures due to other errors	因其它错误失败的次数
Packets out of sequence	报文失序的次数
ICMP-jitter results	ICMP-jitter测试的结果，只在ICMP-jitter测试中存在此信息
Packets arrived late	探测超时后，收到的响应报文个数
UDP-jitter results	UDP-jitter测试的结果，只在UDP-jitter测试中存在此信息
Voice results	Voice测试的结果，只在Voice测试中存在此信息
RTT number	收到的响应报文数
Min positive SD	源到目的方向正抖动时延的最小值
Min positive DS	目的到源方向正抖动时延的最小值
Max positive SD	源到目的方向正抖动时延的最大值
Max positive DS	目的到源方向正抖动时延的最大值
Positive SD number	源到目的方向正抖动时延的数目

字段	描述
Positive DS number	目的到源方向正抖动时延的数目
Positive SD sum	源到目的方向正抖动时延之和
Positive DS sum	目的到源方向正抖动时延之和
Positive SD average	源到目的方向正抖动时延的平均值
Positive DS average	目的到源方向正抖动时延的平均值
Positive SD square-sum	源到目的方向正抖动时延的平方和
Positive DS square-sum	目的到源方向正抖动时延的平方和
Min negative SD	源到目的方向负抖动时延的绝对值的最小值
Min negative DS	目的到源方向负抖动时延的绝对值的最小值
Max negative SD	源到目的方向负抖动时延的绝对值的最大值
Max negative DS	目的到源方向负抖动时延的绝对值的最大值
Negative SD number	源到目的方向负抖动时延的数目
Negative DS number	目的到源方向负抖动时延的数目
Negative SD sum	源到目的方向负抖动时延的绝对值之和
Negative DS sum	目的到源方向负抖动时延的绝对值之和
Negative SD average	源到目的方向负抖动时延的绝对值的平均值
Negative DS average	目的到源方向负抖动时延的绝对值的平均值
Negative SD square-sum	源到目的方向负抖动时延的平方和
Negative DS square-sum	目的到源方向负抖动时延的平方和
One way results	单向延迟测试结果，只有ICMP-jitter、UDP-Jitter和Voice类型测试有单向延迟测试结果
Max SD delay	源到目的的最大时延
Max DS delay	目的到源的最大时延
Min SD delay	源到目的的最小时延
Min DS delay	目的到源的最小时延
Number of SD delay	源到目的计算的时延数
Number of DS delay	目的到源计算的时延数
Sum of SD delay	源到目的的时延和
Sum of DS delay	目的到源的时延和
Square-Sum of SD delay	源到目的的时延的平方和
Square-Sum of DS delay	目的到源的时延的平方和
SD lost packets	源到目的方向丢失的报文个数
DS lost packets	目的到源方向丢失的报文个数

字段	描述
Lost packets for unknown reason	不能确定原因丢失的报文个数
Voice scores	语音参数，只在Voice类型测试有此信息
MOS value	为语音计算的MOS值
ICPIF value	为语音计算的ICPIF值
Hop IP	本跳IP地址，只在Path-jitter测试中存在此信息
Path-jitter results	Path-jitter测试的结果，只在Path-jitter测试中存在此信息
Jitter number	计算抖动次数，只在Path-jitter测试中存在此信息
Min/Max/Average jitter	最小/最大/平均抖动时延，单位为毫秒，只在Path-jitter测试中存在此信息
Positive jitter number	正抖动时延的数目，只在Path-jitter测试中存在此信息
Min/Max/Average positive jitter	最小/最大/平均正抖动时延，单位为毫秒，只在Path-jitter测试中存在此信息
Sum/Square-Sum positive jitter	正抖动时延之和/平方和，只在Path-jitter测试中存在此信息
Negative jitter number	负抖动时延的数目，只在Path-jitter测试中存在此信息
Min/Max/Average negative jitter	最小/最大/平均负抖动时延，单位为毫秒，只在Path-jitter测试中存在此信息
Sum/Square-Sum negative jitter	负抖动时延之和/平方和，只在Path-jitter测试中存在此信息
TTL	本次收到的应答报文中的TTL值
Hop IP	回复应答的节点IP地址
Time	收到应答报文的时间

### 1.1.14 display nqa statistics

**display nqa statistics** 命令用来显示 NQA 测试的统计信息。

#### 【命令】

**display nqa statistics** [ *admin-name operation-tag* ]

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

*admin-name* : 创建 NQA 测试组的管理员名称，为 1～32 个字符的字符串，字符串中不能包括“-”，不区分大小写。

**operation-tag**: 测试操作的标签，为 1~32 个字符的字符串，字符串中不能包括“-”，不区分大小写。

### 【使用指导】

测试开始后，如果第一次测试中的所有探测尚未完成，则无法生成统计信息。若此时通过该命令查看统计信息，则显示信息为全 0。

如果配置了阈值告警组，将显示在 **statistics interval** 命令指定的统计周期内的监测结果。若阈值告警组的阈值类型为平均值，或监测对象为 Voice 测试的 ICPIF 或 MOS 值，则显示的监测结果为无效值。

不指定任何参数，则表示显示所有测试组的统计信息。

### 【举例】

# 显示 NQA 测试组管理员名称为 **admin**、测试操作标签为 **test** 的 TCP 测试的统计信息。

```
<Sysname> display nqa statistics admin test
NQA entry (admin admin, tag test) test statistics:
NO. : 1
  Start time: 2007-01-01 09:30:20.0
  Life time: 2 seconds
  Send operation times: 1          Receive response times: 1
  Min/Max/Average round trip time: 13/13/13
  Square-Sum of round trip time: 169
Extended results:
  Packet loss ratio: 0%
  Failures due to timeout: 0
  Failures due to disconnect: 0
  Failures due to no connection: 0
  Failures due to internal error: 0
  Failures due to other errors: 0
```

# 显示 NQA 测试组管理员名称为 **admin**、测试操作标签为 **test** 的 ICMP-jitter 测试的统计信息。

```
<Sysname> display nqa statistics admin test
NQA entry (admin admin, tag test) test statistics:
NO. : 1
  Start time: 2015-03-09 17:42:10.7
  Life time: 156 seconds
  Send operation times: 1560      Receive response times: 1560
  Min/Max/Average round trip time: 1/2/1
  Square-Sum of round trip time: 1563
Extended results:
  Packet loss ratio: 0%
  Failures due to timeout: 0
  Failures due to internal error: 0
  Failures due to other errors: 0
  Packets out of sequence: 0
  Packets arrived late: 0
ICMP-jitter results:
  RTT number: 1560
  Min positive SD: 1              Min positive DS: 1
```

```

Max positive SD: 1
Positive SD number: 18
Positive SD sum: 18
Positive SD average: 1
Positive SD square-sum: 18
Min negative SD: 1
Max negative SD: 1
Negative SD number: 24
Negative SD sum: 24
Negative SD average: 1
Negative SD square-sum: 24

Max positive DS: 2
Positive DS number: 46
Positive DS sum: 49
Positive DS average: 1
Positive DS square-sum: 55
Min negative DS: 1
Max negative DS: 2
Negative DS number: 57
Negative DS sum: 58
Negative DS average: 1
Negative DS square-sum: 60

One way results:
Max SD delay: 1
Min SD delay: 1
Number of SD delay: 4
Sum of SD delay: 4
Square-Sum of SD delay: 4
Lost packets for unknown reason: 0

Max DS delay: 2
Min DS delay: 1
Number of DS delay: 4
Sum of DS delay: 5
Square-Sum of DS delay: 7

```

Reaction statistics:

Index	Checked Element	Threshold Type	Checked Num	Over-threshold Num
1	jitter-DS	accumulate	1500	10
2	jitter-SD	average	-	-
3	OWD-DS	-	1560	2
4	OWD-SD	-	1560	0
5	packet-loss	accumulate	0	0
6	RTT	accumulate	1560	0

# 显示 NQA 测试组管理员名称为 **admin**、测试操作标签为 **test** 的 UDP-jitter 测试的统计信息。

```
<Sysname> display nqa statistics admin test
```

NQA entry (admin admin, tag test) test statistics:

NO. : 1

Start time: 2007-01-01 09:33:22.3

Life time: 23 seconds

Send operation times: 100

Receive response times: 100

Min/Max/Average round trip time: 1/11/5

Square-Sum of round trip time: 24360

Extended results:

Packet loss ratio: 0%

Failures due to timeout: 0

Failures due to internal error: 0

Failures due to other errors: 0

Packets out of sequence: 0

Packets arrived late: 0

UDP-jitter results:

RTT number: 550

Min positive SD: 1

Min positive DS: 1

Max positive SD: 7

Max positive DS: 1

Positive SD number: 220

Positive DS number: 97

Positive SD sum: 283

Positive DS sum: 287

```

Positive SD average: 1
Positive SD square-sum: 709
Min negative SD: 2
Max negative SD: 10
Negative SD number: 81
Negative SD sum: 556
Negative SD average: 6
Negative SD square-sum: 4292

Positive DS average: 2
Positive DS square-sum: 1937
Min negative DS: 1
Max negative DS: 1
Negative DS number: 94
Negative DS sum: 191
Negative DS average: 2
Negative DS square-sum: 967

One way results:
Max SD delay: 5
Min SD delay: 1
Number of SD delay: 550
Sum of SD delay: 1475
Square-Sum of SD delay: 5407
SD lost packets: 0
Lost packets for unknown reason: 0

Max DS delay: 5
Min DS delay: 1
Number of DS delay: 550
Sum of DS delay: 1201
Square-Sum of DS delay: 3959
DS lost packets: 0

```

Reaction statistics:

Index	Checked Element	Threshold Type	Checked Num	Over-threshold Num
1	jitter-DS	accumulate	90	25
2	jitter-SD	average	-	-
3	OWD-DS	-	100	24
4	OWD-SD	-	100	13
5	packet-loss	accumulate	0	0
6	RTT	accumulate	100	52

# 显示 NQA 测试组管理员名称为 **admin**、测试操作标签为 **test** 的 Voice 测试的统计信息。

<Sysname> display nqa statistics admin test

NQA entry (admin admin, tag test) test statistics:

NO. : 1

Start time: 2007-01-01 09:33:45.3

Life time: 120 seconds

Send operation times: 10

Receive response times: 10

Min/Max/Average round trip time: 1/12/7

Square-Sum of round trip time: 620

Extended results:

Packet loss ratio: 0%

Failures due to timeout: 0

Failures due to internal error: 0

Failures due to other errors: 0

Packets out of sequence: 0

Packets arrived late: 0

Voice results:

RTT number: 10

Min positive SD: 3

Min positive DS: 1

Max positive SD: 10

Max positive DS: 1

Positive SD number: 3

Positive DS number: 2

Positive SD sum: 18

Positive DS sum: 2

Positive SD average: 6

Positive DS average: 1

Positive SD square-sum: 134

Positive DS square-sum: 2



```

Min negative SD: 3
Max negative SD: 9
Negative SD number: 4
Negative SD sum: 25
Negative SD average: 6
Negative SD square-sum: 187
Min negative DS: 1
Max negative DS: 1
Negative DS number: 2
Negative DS sum: 2
Negative DS average: 1
Negative DS square-sum: 2
One way results:
Max SD delay: 0
Min SD delay: 0
Number of SD delay: 0
Sum of SD delay: 0
Square-Sum of SD delay: 0
SD lost packets: 0
Lost packets for unknown reason: 0
Max DS delay: 0
Min DS delay: 0
Number of DS delay: 0
Sum of DS delay: 0
Square-Sum of DS delay: 0
DS lost packets: 0
Voice scores:
Max MOS value: 4.40
Min MOS value: 4.40
Max ICPIF value: 0
Min ICPIF value: 0
Reaction statistics:
  Index  Checked Element  Threshold Type  Checked Num  Over-threshold Num
  1      ICPIF           -              -             -
  2      MOS            -              -             -
# 显示 NQA 测试组管理员名称为 admin、测试操作标签为 test 的 Path-jitter 测试的统计信息。
<Sysname> display nqa statistics admin test
NQA entry (admin admin, tag test) test statistics:
NO. : 1
Path 1:
Hop IP 192.168.40.210
Basic Results:
  Send operation times: 10
  Receive response times: 10
  Min/Max/Average round trip time: 1/1/1
  Square-Sum of round trip time: 10
Extended Results:
  Packet loss ratio: 0%
  Failures due to timeout: 0
  Failures due to internal error: 0
  Failures due to other errors: 0
  Packets out of sequence: 0
  Packets arrived late: 0
Path-Jitter Results:
  Jitter number: 9
  Min/Max/Average jitter: 0/0/0
  Positive jitter number: 0
  Min/Max/Average positive jitter: 0/0/0
  Sum/Square-Sum positive jitter: 0/0
  Negative jitter number: 0
  Min/Max/Average negative jitter: 0/0/0
  Sum/Square-Sum negative jitter: 0/0

```

Hop IP 192.168.50.209

Basic Results:

Send operation times: 10  
Receive response times: 10  
Min/Max/Average round trip time: 1/1/1  
Square-Sum of round trip time: 10

Extended Results:

Packet loss ratio: 0%  
Failures due to timeout: 0  
Failures due to internal error: 0  
Failures due to other errors: 0  
Packets out of sequence: 0  
Packets arrived late: 0

Path-Jitter Results:

Jitter number: 9  
Min/Max/Average jitter: 0/0/0  
Positive jitter number: 0  
Min/Max/Average positive jitter: 0/0/0  
Sum/Square-Sum positive jitter: 0/0  
Negative jitter number: 0  
Min/Max/Average negative jitter: 0/0/0  
Sum/Square-Sum negative jitter: 0/0

表1-7 display nqa statistics 命令显示信息描述

字段	描述
No.	统计组的组号
Start time	测试组启动时间
Life time	测试的持续时间，单位为秒
Send operation times	发送的探测报文数
Receive response times	收到的响应报文数
Min/Max/Average round trip time	最小/最大/平均往返时间，单位为毫秒
Square-Sum of round trip time	往返时间平方和
Packet loss ratio	平均丢包率
Failures due to timeout	测试过程中超时的次数
Failures due to disconnect	对方强制断开连接的次数
Failures due to no connection	和对方建立连接失败的次数
Failures due to internal error	因内部错误失败的次数
Failures due to other errors	因其它错误失败的次数
Packets out of sequence	报文失序的次数
Packets arrived late	迟到报文个数
ICMP-jitter results	ICMP-jitter测试的结果，只在ICMP-jitter测试中存在此信息

字段	描述
UDP-jitter results	UDP-jitter测试的结果，只在UDP-jitter测试中存在此信息
Voice results	Voice测试的结果，只在Voice测试中存在此信息
RTT number	收到的响应报文数
Min positive SD	源到目的方向抖动时延为正值的的最小值
Min positive DS	目的到源方向抖动时延为正值的的最小值
Max positive SD	源到目的方向抖动时延为正值的最大值
Max positive DS	目的到源方向抖动时延为正值的最大值
Positive SD number	源到目的方向抖动时延为正值的数目
Positive DS number	目的到源方向抖动时延为正值的数目
Positive SD sum	源到目的方向抖动时延为正值的和
Positive DS sum	目的到源方向抖动时延为正值的和
Positive SD average	源到目的方向抖动时延为正值的平均值
Positive DS average	目的到源方向抖动时延为正值的平均值
Positive SD square-sum	源到目的方向抖动时延为正值的平方和
Positive DS square-sum	目的到源方向抖动时延为正值的平方和
Min negative SD	源到目的方向抖动时延为负值的最小绝对值
Min negative DS	目的到源方向抖动时延为负值的最小绝对值
Max negative SD	源到目的方向抖动时延为负值的最大绝对值
Max negative DS	目的到源方向抖动时延为负值的最大绝对值
Negative SD number	源到目的方向抖动时延为负值的数目
Negative DS number	目的到源方向抖动时延为负值的数目
Negative SD sum	源到目的方向抖动时延为负值的绝对值和
Negative DS sum	目的到源方向抖动时延为负值的绝对值和
Negative SD average	源到目的方向抖动时延为负值的绝对值的平均值
Negative DS average	目的到源方向抖动时延为负值的绝对值的平均值
Negative SD square-sum	源到目的方向抖动时延为负值的平方和
Negative DS square-sum	目的到源方向抖动时延为负值的平方和
One way results	单向延迟测试结果，只有ICMP-jitter、UDP-Jitter和Voice类型测试有单向延迟测试结果
Max SD delay	源到目的的最大时延
Max DS delay	目的到源的最大时延
Min SD delay	源到目的的最小时延
Min DS delay	目的到源的最小时延

字段	描述
Number of SD delay	源到目的计算的时延数
Number of DS delay	目的到源计算的时延数
Sum of SD delay	源到目的的时延和
Sum of DS delay	目的到源的时延和
Square-Sum of SD delay	源到目的的时延的平方和
Square-Sum of DS delay	目的到源的时延的平方和
SD lost packets	源到目的方向丢失的报文个数
DS lost packets	目的到源方向丢失的报文个数
Lost packets for unknown reason	不能确定原因丢失的报文个数
Voice scores	语音参数，只在voice类型测试有此信息
Max MOS value	最大MOS值
Min MOS value	最小MOS值
Max ICPIF value	最大ICPIF值
Min ICPIF value	最小ICPIF值
Reaction statistics	阈值告警组在统计周期内的监测结果
Index	阈值告警组的编号
Checked Element	监测对象
Threshold Type	阈值类型
Checked Num	已监测的样本个数
Over-threshold Num	超出阈值的样本个数
Path	Path-jitter测试结果的路径序号，只在Path-jitter测试中存在此信息
Hop IP	本跳IP地址，只在Path-jitter测试中存在此信息
Path-jitter results	Path-jitter测试的结果，只在Path-jitter测试中存在此信息
Jitter number	计算抖动次数，只在Path-jitter测试中存在此信息
Min/Max/Average jitter	最小/最大/平均抖动时延，单位为毫秒，只在Path-jitter测试中存在此信息
Positive jitter number	正抖动时延的数目，只在Path-jitter测试中存在此信息
Min/Max/Average positive jitter	最小/最大/平均正抖动时延，单位为毫秒，只在Path-jitter测试中存在此信息
Sum/Square-Sum positive jitter	正抖动时延之和/平方和，只在Path-jitter测试中存在此信息
Negative jitter number	负抖动时延的数目，只在Path-jitter测试中存在此信息
Min/Max/Average negative jitter	最小/最大/平均负抖动时延，单位为毫秒，只在Path-jitter测试中存在此信息
Sum/Square-Sum negative jitter	负抖动时延之和/平方和，只在Path-jitter测试中存在此信息

表1-8 display nqa statistics 命令显示阈值告警功能相关字段取值描述  
(DHCP/DLSw/DNS/FTP/HTTP/ICMP-echo/SNMP/TCP/UDP-echo 测试类型)

监测对象	阈值类型	监测的样本范围	Checked Num 取值	Over-threshold Num 取值
probe-duration	accumulate	统计周期内, 进行的探测	统计周期内, 已完成的探测次数	统计周期内, 累计的探测持续时间不在阈值范围内的探测次数
	average	-	-	-
	consecutive	统计周期内, 进行的探测	统计周期内, 已完成的探测次数	统计周期内, 连续的探测持续时间不在阈值范围内的探测次数
probe-fail	accumulate	统计周期内, 进行的探测	统计周期内, 已完成的探测次数	统计周期内, 累计的失败的探测次数
	consecutive	统计周期内, 进行的探测	统计周期内, 已完成的探测次数	统计周期内, 连续的失败的探测次数

表1-9 display nqa statistics 命令显示阈值告警功能相关字段取值描述 (ICMP-jitter/UDP-jitter/Voice 测试类型)

监测对象	阈值类型	监测的样本范围	Checked Num 取值	Over-threshold Num 取值
RTT	accumulate	统计周期内, 发送的报文	统计周期内, 已发送的报文个数	统计周期内, 累计的往返时间不在阈值范围内的报文个数
	average	-	-	-
jitter-DS/jitter-SD	accumulate	统计周期内, 发送的报文	统计周期内, 已发送的报文个数	统计周期内, 累计的单向抖动时间不在阈值范围内的报文个数
	average	-	-	-
OWD-DS/OWD-SD	-	统计周期内, 发送的报文	统计周期内, 已发送的报文个数	统计周期内, 单向时延不在阈值范围内的报文个数
packet-loss	accumulate	统计周期内, 发送的报文	统计周期内, 已发送的报文个数	统计周期内累计的丢包数
ICPIF/MOS (仅 Voice 测试支持)	-	-	-	-

## 【相关命令】

- `statistics interval`

### 1.1.15 expect data

`expect data` 命令用来配置期望的应答内容。

`undo expect data` 命令用来恢复缺省情况。

## 【命令】

```
expect data expression [ offset number ]  
undo expect data
```

## 【缺省情况】

未配置期望的应答内容。

## 【视图】

HTTP/HTTPS/TCP/UDP 类型的 NQA 模板视图

## 【缺省用户角色】

network-admin

## 【参数】

**expression**: 期望收到的应答内容，为 1~200 个字符的字符串，区分大小写。

**offset number**: 所期望的内容在返回报文中的偏移量，取值范围为 0~1000，缺省值为 0。如果不指定本参数，则设备直接从返回报文的第一个字节开始匹配。

## 【使用指导】

NQA 客户端对于返回报文中期望的应答内容匹配方式如下：

- 如果未配置 **offset** 参数，则设备直接从返回报文的第一个字节开始匹配，若不匹配，继续从第二个字节开始匹配，以此类推；
- 如果配置了 **offset** 参数，则设备从返回报文偏移量之后的第一个字节开始匹配，若匹配失败，则忽略该偏移量，从返回报文的第一个字节开始匹配，若不匹配，继续从第二个字节开始匹配，以此类推。

无论使用以上哪种匹配方式，只要返回报文中包含期望收到的应答内容，则表示当前 NQA 目的端设备合法；否则为非法设备。

对于 HTTP/HTTPS 类型的 NQA 模板，仅当回应报文中存在 Content-Length 头域时，进行期望应答内容的检查，否则不做检查。

对于 TCP 类型的 NQA 模板，仅当 **data-fill** 和 **expect data** 命令都配置时，进行期望应答内容的检查，否则不做检查。

对于 UDP 类型的 NQA 模板，：

- 仅当 **data-fill** 和 **expect data** 命令都配置时，进行期望应答内容的检查，否则不做检查。
- 由于 UDP 报文数据字段的前 5 个字节具有特定用途。缺省情况下，配置 **expect data** 后从第 6 个字节开始进行偏移检查。

## 【举例】

# 在 HTTP 类型的 NQA 模板视图下配置期望的应答内容为 welcome!。

```
<Sysname> system-view  
[Sysname] nqa template http httptplt  
[Sysname-nqatplt-http-httptplt] expect data welcome!
```

### 1.1.16 expect ip

**expect ip** 命令用来配置期望返回的 IP 地址。

**undo expect ip** 命令用来恢复缺省情况。

#### 【命令】

**expect ip ip-address**

**undo expect ip**

#### 【缺省情况】

未配置期望返回的 IP 地址。

#### 【视图】

DNS 类型的 NQA 模板视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**ip-address**: DNS 探测期望返回的 IP 地址。

#### 【使用指导】

在 DNS 测试中, NQA 客户端通过该命令配置的 IP 地址与 DNS 服务器通过域名解析出的 IP 地址进行比较, 若相同, 则证明目前测试的 DNS 服务器合法, 否则为非法 DNS 服务器。

#### 【举例】

# 在 DNS 类型的 NQA 模板视图下配置期望返回的 IP 地址为 1.1.1.1。

```
<Sysname> system-view
[Sysname] nqa template dns dnstplt
[Sysname-nqatplt-dns-dnstplt] expect ip 1.1.1.1
```

### 1.1.17 expect ipv6

**expect ipv6** 命令用来配置期望返回的 IPv6 地址。

**undo expect ipv6** 命令用来恢复缺省情况。

#### 【命令】

**expect ipv6 ipv6-address**

**undo expect ipv6**

#### 【缺省情况】

未配置期望返回的 IPv6 地址。

#### 【视图】

DNS 类型的 NQA 模板视图

#### 【缺省用户角色】

network-admin

### 【参数】

*ipv6-address*: DNS 探测期望返回的 IPv6 地址。

### 【使用指导】

在 DNS 测试中，NQA 客户端通过该命令配置的 IPv6 地址与 DNS 服务器通过域名解析出的 IPv6 地址进行比较，若相同，则证明目前测试的 DNS 服务器合法，否则为非法 DNS 服务器。

### 【举例】

# 在 DNS 类型的 NQA 模板视图下配置期望返回的 IPv6 地址为 1::1。

```
<Sysname> system-view
[Sysname] nqa template dns dnstplt
[Sysname-nqatplt-dns-dnstplt] expect ipv6 1::1
```

## 1.1.18 expect status

**expect status** 命令用来配置期望的应答状态码。

**undo expect status** 命令用来恢复缺省情况。

### 【命令】

```
expect status status-list
undo expect status [status-list]
```

### 【缺省情况】

未配置期望的应答状态码。

### 【视图】

HTTP/HTTPS 类型的模板视图

### 【缺省用户角色】

network-admin

### 【参数】

*status-list*: 状态码列表，即 HTTP/HTTPS 模板类型期望收到的状态码范围。表示方式为 *status-list* = { *status-num1* [ *to status-num2* ] } &<1-10>, *status-num* 取值范围为 0~999, *status-num2* 的值要大于或等于 *status-num1* 的值, &<1-10> 表示前面的参数最多可以重复输入 10 次。

### 【使用指导】

HTTP/HTTPS 类型的 NQA 模板支持配置状态码。报文的状态码是由 3 位十进制数组成的字段，它包含服务器的状态信息，用户可以根据该状态码了解服务器的状态。状态码的第一位规定状态码的类型。

### 【举例】

# 在 HTTP 类型的 NQA 模板视图下配置期望的应答状态码，允许状态码为 200、300、400~500。

```
<Sysname> system-view
[Sysname] nqa template http httptplt
[Sysname-nqatplt-http-httptplt] expect status 200 300 400 to 500
```



### 1.1.19 filename

**filename** 命令用来配置 FTP 服务器和客户端之间传送文件的文件名。

**undo filename** 命令用来恢复缺省情况。

#### 【命令】

**filename** *filename*

**undo filename**

#### 【缺省情况】

未配置 FTP 服务器和客户端之间传送文件的文件名。

#### 【视图】

FTP 测试类型视图

FTP 类型的 NQA 模板视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**filename**: FTP 服务器和客户端之间传送文件的文件名，为 1~200 个字符的字符串，字符串中不能包括“/”，区分大小写。

#### 【举例】

# 在 FTP 测试类型下配置 FTP 服务器和客户端之间要传送文件的文件名为 config.txt。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type ftp
[Sysname-nqa-admin-test-ftp] filename config.txt
```

# 在 FTP 类型的 NQA 模板视图下配置 FTP 服务器和客户端之间要传送文件的文件名为 config.txt。

```
<Sysname> system-view
[Sysname] nqa template ftp ftptplt
[Sysname-nqatplt-ftp-ftptplt] filename config.txt
```

### 1.1.20 frequency

**frequency** 命令用来配置测试组连续两次测试开始时间的时间间隔。

**undo frequency** 命令用来恢复缺省情况。

#### 【命令】

**frequency** *interval*

**undo frequency**

#### 【缺省情况】

在 NQA 测试类型视图下，Voice、Path-jitter 测试中连续两次测试开始时间的时间间隔为 60000 毫秒；其他类型的测试中连续两次测试开始时间的时间间隔为 0 毫秒，即只进行一次测试。

在 NQA 模板视图下，测试中连续两次测试开始时间的时间间隔为 5000 毫秒。

### 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图  
DHCP/DLSw/DNS/FTP/HTTP/SNMP 测试类型视图  
UDP-tracert 测试类型视图  
ICMP-jitter/Path-jitter/UDP-jitter/Voice 测试类型视图  
任意类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

*interval*: 连续两次测试开始时间的时间间隔，取值范围为 0~604800000，单位为毫秒。时间间隔为 0，表示两次测试的时间间隔为无穷，即只进行一次测试，此时不会生成统计结果。

### 【使用指导】

通过 **nqa schedule** 命令启动 NQA 测试组后，每隔 *interval* 时间启动一次测试。  
需要注意的是，如果到达 **frequency** 指定的时间间隔时，上次测试尚未完成，则不启动新一轮测试。

### 【举例】

```
# 在 ICMP-echo 测试类型下配置连续两次测试开始时间的的时间间隔为 1000 毫秒。
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] frequency 1000

# 在 DNS 类型的 NQA 模板视图下配置连续两次测试开始时间的的时间间隔为 1000 毫秒。
<Sysname> system-view
[Sysname] nqa template dns dnstplt
[Sysname-nqatplt-dns-dnstplt] frequency 1000
```

## 1.1.21 history-record enable

**history-record enable** 命令用来开启 NQA 测试组的历史记录保存功能。  
**undo history-record enable** 命令用来关闭 NQA 测试组的历史记录保存功能。

### 【命令】

```
history-record enable
undo history-record enable
```

### 【缺省情况】

UDP-tracert 测试类型的历史记录保存功能处于开启状态，其他类型的 NQA 测试组的历史记录保存功能处于关闭状态。

### 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图  
DHCP/DLSw/DNS/FTP/HTTP/SNMP 测试类型视图

UDP-tracert 测试类型视图

【缺省用户角色】

network-admin

【使用指导】

如果开启 NQA 测试组的历史记录保存功能，则系统会记录该 NQA 测试组的历史信息，通过 **display nqa history** 命令可以查看该测试组的历史记录信息。

如果关闭 NQA 测试组的历史记录保存功能，则系统不会记录该测试组的历史信息，原有的历史记录信息也会被删除。

【举例】

# 在 ICMP-echo 测试类型下开启 NQA 测试组的历史记录保存功能。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] history-record enable
```

【相关命令】

- **display nqa history**

## 1.1.22 history-record keep-time

**history-record keep-time** 命令用来配置 NQA 测试组中历史记录的保存时间。

**undo history-record keep-time** 命令用来恢复缺省情况。

【命令】

```
history-record keep-time keep-time
undo history-record keep-time
```

【缺省情况】

NQA 测试组中历史记录的保存时间为 120 分钟。

【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图

DHCP/DLSw/DNS/FTP/HTTP/SNMP 测试类型视图

UDP-tracert 测试类型视图

【缺省用户角色】

network-admin

【参数】

*keep-time*：历史记录的保存时间，取值范围为 1~1440，单位为分钟。

【使用指导】

NQA 测试结束后，开始计算该测试组中所有历史记录的保存时间。保存时间达到配置的值后，将删除这些记录。

### 【举例】

# 在 ICMP-echo 测试类型下配置 NQA 测试组中历史记录的保存时间为 100 分钟。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] history-record keep-time 100
```

### 1.1.23 history-record number

**history-record number** 命令用来配置在一个测试组中能够保存的最大历史记录个数。

**undo history-record number** 命令用来恢复缺省情况。

### 【命令】

```
history-record number number
undo history-record number
```

### 【缺省情况】

一个测试组中能够保存的最大历史记录个数为 50。

### 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图

DHCP/DLSw/DNS/FTP/HTTP/SNMP 测试类型视图

UDP-tracert 测试类型视图

### 【缺省用户角色】

network-admin

### 【参数】

**number**: 在一个测试组中能够保存的最大历史记录个数，取值范围为 0~50。

### 【使用指导】

如果一个测试组中历史记录个数超过设定的最大数目，则最早的历史记录将会被删除。

### 【举例】

# 在 ICMP-echo 测试类型下配置一个测试组中能够保存的最大历史记录数为 10 个。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] history-record number 10
```

### 1.1.24 init-ttl

**init-ttl** 命令用来配置 UDP-tracert 探测报文的初始跳数。

**undo init-ttl** 命令用来恢复缺省情况。

### 【命令】

```
init-ttl value
undo init-ttl
```

### 【缺省情况】

UDP-tracert 探测报文的初始跳数为 1。

### 【视图】

UDP-tracert 测试类型视图

### 【缺省用户角色】

network-admin

### 【参数】

**value:** UDP-tracert 探测报文的初始跳数，取值范围 1～255。

### 【举例】

# 在 UDP-tracert 测试类型下配置 UDP-tracert 探测报文的初始跳数为 5 跳。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type udp-tracert
[Sysname-nqa-admin-test-udp-tracert] init-ttl 5
```

## 1.1.25 key

**key** 命令用来设置 RADIUS 认证使用的共享密钥。

**undo key** 命令用来恢复缺省情况。

### 【命令】

```
key { cipher | simple } string
undo key
```

### 【缺省情况】

未设置 RADIUS 认证使用的共享密钥。

### 【视图】

RADIUS 类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

**cipher:** 以密文方式设置密钥。

**simple:** 以明文方式设置密钥，该密钥将以密文形式存储。

**string:** 密钥字符串，区分大小写。明文密钥为 1～64 个字符的字符串，密文密钥为 1～117 个字符的字符串。

### 【使用指导】

必须保证设备上设置的共享密钥与 RADIUS 服务器上的完全一致。

### 【举例】

# 在 RADIUS 类型的 NQA 模板视图下设置 RADIUS 认证使用的共享密钥为明文 abc。

```
<Sysname> system-view
[Sysname] nqa template radius radiustplt
[Sysname-nqatplt-radius-radiustplt] key simple abc
```

### 1.1.26 lsr-path

**lsr-path** 命令用来配置松散路由。

**undo lsr-path** 命令用来恢复缺省情况。

#### 【命令】

```
lsr-path ip-address&<1-8>
undo lsr-path
```

#### 【缺省情况】

未配置松散路由。

#### 【视图】

Path-jitter 测试类型视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**ip-address&<1-8>**: 松散路由 IP 地址，&<1-8>表示最多可以输入 8 个 IP 地址，每个 IP 地址之间用空格分隔。

#### 【使用指导】

通过本命令配置松散路由，用户只需给出 NQA 探测报文必须经过的一些“节点”，并不需要给出一条完备的路径，无直接连接的“节点”之间的路由需要路由器寻址功能补充。

Path-jitter 测试中，NQA 客户端通过 **tracert** 过程使用该命令配置的松散路由进行探路，并根据收到 ICMP 报文计算主要“节点”时延和抖动时间。

#### 【举例】

# 在 Path-jitter 测试类型下配置松散路由为 10.1.1.20 和 10.1.2.10 两跳。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type path-jitter
[Sysname-nqa-admin-test- path-jitter] lsr-path 10.1.1.20 10.1.2.10
```

### 1.1.27 max-failure

**max-failure** 命令用来配置一次 UDP-tracert 测试中连续探测失败的最大次数。

**undo max-failure** 命令用来恢复缺省情况。

#### 【命令】

```
max-failure times
undo max-failure
```

### 【缺省情况】

一次 UDP-tracert 测试中连续探测失败的最大次数为 5。

### 【视图】

UDP-tracert 测试类型视图

### 【缺省用户角色】

network-admin

### 【参数】

**times:** 表示一次 UDP-tracert 测试中连续探测失败的最大次数，取值范围 0~255。0 和 255 意味着 UDP-tracert 探测不会因为连续探测失败而停止测试。

### 【使用指导】

当 UDP-tracert 测试中连续探测失败的次数达到指定的最大值时，设备认为测试失败并停止探测。

### 【举例】

# 在 UDP-tracert 测试类型下配置一次 UDP-tracert 测试中连续探测失败的最大次数为 20 次。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type udp-tracert
[Sysname-nqa-admin-test-udp-tracert] max-failure 20
```

## 1.1.28 mode

**mode** 命令用来配置 FTP 测试的数据传输方式。

**undo mode** 命令用来恢复缺省情况。

### 【命令】

```
mode { active | passive }
undo mode
```

### 【缺省情况】

FTP 测试的数据传输方式为主动方式。

### 【视图】

FTP 测试类型视图

FTP 类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

**active:** 设置 FTP 的数据传输方式为主动方式。表示在建立数据连接时由服务器主动发起连接请求。

**passive:** 设置 FTP 的数据传输方式为被动方式。表示在建立数据连接时由客户端主动发起连接请求。

### 【举例】

# 在 FTP 测试类型下配置 FTP 测试的数据传输方式为被动方式。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type ftp
[Sysname-nqa-admin-test-ftp] mode passive
```

# 在 FTP 类型的 NQA 模板视图下配置数据传输方式为被动方式。

```
<Sysname> system-view
[Sysname] nqa template ftp ftptplt
[Sysname-nqatplt-ftp-ftptplt] mode passive
```

## 1.1.29 next-hop ip

**next-hop ip** 命令用来配置探测报文的下一跳 IP 地址。

**undo next-hop ip** 命令用来恢复缺省情况。

### 【命令】

```
next-hop ip ip-address
undo next-hop ip
```

### 【缺省情况】

未配置探测报文的下一跳 IP 地址。

### 【视图】

ICMP-echo 测试类型视图

ICMP/TCP Half Open 类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

*ip-address*: 探测报文的下一跳 IP 地址。

### 【使用指导】

配置本命令之后，设备按照指定的下一跳地址发送探测报文；若未配置本命令，则按照路由表缺省的转发规则发送探测报文。

### 【举例】

# 在 ICMP-echo 测试类型下配置探测报文的下一跳 IP 地址为 10.1.1.1。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] next-hop ip 10.1.1.1
```

## 1.1.30 next-hop ipv6

**next-hop ipv6** 命令用来配置探测报文的下一跳 IPv6 地址。

**undo next-hop ipv6** 命令用来恢复缺省情况。



### 【命令】

```
next-hop ipv6 ipv6-address  
undo next-hop ipv6
```

### 【缺省情况】

未配置探测报文的下一跳 IPv6 地址。

### 【视图】

ICMP-echo 测试类型视图

ICMP/TCP Half Open 类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

*ipv6-address*: 探测报文的下一跳 IPv6 地址，不支持 IPv6 链路本地地址。

### 【使用指导】

配置本命令之后，设备按照指定的下一跳地址发送探测报文；若未配置本命令，则按照路由表缺省的转发规则发送探测报文。

### 【举例】

# 在 ICMP-echo 测试类型下配置探测报文的下一跳 IPv6 地址为 10::1。

```
<Sysname> system-view  
[Sysname] nqa entry admin test  
[Sysname-nqa-admin-test] type icmp-echo  
[Sysname-nqa-admin-test-icmp-echo] next-hop ipv6 10::1
```

## 1.1.31 no-fragment enable

**no-fragment enable** 命令用来开启 UDP-tracert 探测类型的禁止报文分片功能。

**undo no-fragment enable** 命令用来关闭 UDP-tracert 探测类型的禁止报文分片功能。

### 【命令】

```
no-fragment enable  
undo no-fragment enable
```

### 【缺省情况】

UDP-tracert 测试类型的禁止报文分片功能处于关闭状态。

### 【视图】

UDP-tracert 测试类型视图

### 【缺省用户角色】

network-admin

### 【使用指导】

开启此功能后，设备发送的 IP 报文头部的 DF（don't fragment）字段会被置一，这样报文在转发过程中将无法被分片。通过配置这条命令可以对一条链路的路径 MTU 值进行测试。

### 【举例】

# 在 UDP-tracert 测试类型下开启 UDP-tracert 探测类型的禁止报文分片功能。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type udp-tracert
[Sysname-nqa-admin-test-udp-tracert] no-fragment enable
```

## 1.1.32 nqa

**nqa** 命令用来创建 NQA 测试组，并进入 NQA 测试组视图。如果指定的 NQA 测试组视图已经存在，则直接进入 NQA 测试组视图。

**undo nqa** 命令用来删除指定的 NQA 测试组。

### 【命令】

```
nqa entry admin-name operation-tag
undo nqa { all | entry admin-name operation-tag }
```

### 【缺省情况】

不存在 NQA 测试组。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**admin-name**: 创建 NQA 测试组的管理员名称，为 1~32 个字符的字符串，字符串中不能包括“-”，不区分大小写。

**operation-tag**: 测试操作的标签，为 1~32 个字符的字符串，字符串中不能包括“-”，不区分大小写。

**all**: 所有 NQA 测试组。

### 【举例】

# 创建一个管理员名为 **admin**，测试操作标签为 **test** 的 NQA 测试组，并进入 NQA 测试组视图。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test]
```

## 1.1.33 nqa agent enable

**nqa agent enable** 命令用来开启 NQA 客户端功能。

**undo nqa agent enable** 命令用来关闭 NQA 客户端功能，并停止所有正在进行的测试。

### 【命令】

```
nqa agent enable
undo nqa agent enable
```

### 【缺省情况】

NQA 客户端功能处于开启状态。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【举例】

```
# 开启 NQA 客户端功能。
<Sysname> system-view
[Sysname] nqa agent enable
```

### 【相关命令】

- **nqa server enable**

## 1.1.34 nqa schedule

**nqa schedule** 命令用来配置测试组的启动时间和持续时间。

**undo nqa schedule** 命令用来停止指定测试组的测试。

### 【命令】

```
nqa schedule admin-name operation-tag start-time { hh:mm:ss [ yyyy/mm/dd | mm/dd/yyyy ] | now } lifetime { lifetime | forever } [ recurring ]
undo nqa schedule admin-name operation-tag
```

### 【缺省情况】

未配置 NQA 调度功能。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**admin-name**: 创建 NQA 测试组的管理员名称, 为 1~32 个字符的字符串, 字符串中不能包括“-”, 不区分大小写。

**operation-tag**: 测试操作的标签, 为 1~32 个字符的字符串, 字符串中不能包括“-”, 不区分大小写。

**start-time**: 指定测试组的启动时间和日期。

**hh:mm:ss**: 测试组的启动时间, 小时:分钟:秒。

`yyyy/mm/dd`: 测试组的启动日期, 年:月:日, 缺省值为系统的当前日期, 年的取值范围为 2000~2035。

`mm/dd/yyyy`: 测试组的启动日期, 月:日:年, 缺省值为系统的当前日期, 年的取值范围为 2000~2035。

**now**: 测试组立即开始测试。

**lifetime**: 指定测试的持续时间。

**lifetime**: 测试的持续时间, 取值范围为 1~2147483647, 单位为秒。

**forever**: 测试组将一直进行测试。

**recurring**: 指定测试组每天都被调度运行。每天启动测试的时间由 **start-time** 参数指定。若不指定该参数, 则测试组只在当天的指定时间内调度运行一次。

### 【使用指导】

系统时间在启动时间~启动时间+持续时间范围内时, 测试组进行测试。执行 **nqa schedule** 命令时, 如果系统时间尚未到达启动时间, 则到达启动时间后, 启动测试; 如果系统时间在启动时间~启动时间+持续时间之间, 则立即启动测试; 如果系统时间已经超过启动时间+持续时间, 则不会启动测试。通过 **display clock** 命令可以显示系统的当前时间。

测试组被调度后不允许进入测试类型视图。若需要重新进入测试类型视图, 请使用 **undo nqa schedule** 命令停止指定测试组的测试。

### 【举例】

# 启动管理员名称为 **admin**, 标签为 **test** 的测试组进行测试, 测试组的启动时间为 2008 年 8 月 8 日以后 (包含当天) 的每天的 08:08:08, 测试持续时间为 1000 秒。

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

```
[Sysname] nqa schedule admin test start-time 08:08:08 2008/08/08 lifetime 1000 recurring
```

### 【相关命令】

- **destination ip**
- **display clock** (基础配置命令参考/设备管理)
- **nqa entry**
- **type**

## 1.1.35 nqa template

**nqa template** 命令用来创建指定类型 NQA 模板, 并进入 NQA 模板视图。如果指定的 NQA 模板视图已经存在, 则直接进入 NQA 模板视图。

**undo nqa template** 命令用来删除指定类型的 NQA 模板。

### 【命令】

```
nqa template { dns | ftp | http | https | icmp | radius | ssl | tcp | tcphalfopen |  
udp } name
```

```
undo nqa template { dns | ftp | http | https | icmp | radius | ssl | tcp |  
tcphalfopen | udp } name
```

### 【缺省情况】

不存在 NQA 模板。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**dns**: 配置 DNS 模板类型。

**ftp**: 配置 FTP 模板类型。

**http**: 配置 HTTP 模板类型。

**https**: 配置 HTTPS 模板类型。

**icmp**: 配置 ICMP 模板类型。

**radius**: 配置 RADIUS 模板类型。

**ssl**: 配置 SSL 模板类型。

**tcp**: 配置 TCP 模板类型。

**tcphalfopen**: 配置 TCP Half Open 模板类型。

**udp**: 配置 UDP 模板类型。

**name**: NQA 模板名称，为 1~32 个字符的字符串，不区分大小写。

### 【举例】

# 创建一个类型为 ICMP、名称为 icmptplt 的模板，并进入 NQA 模板视图。

```
<Sysname> system-view
[Sysname] nqa template icmp icmptplt
[Sysname-nqatplt-icmp-icmptplt]
```

## 1.1.36 operation (FTP test type view)

**operation** 命令用来配置 FTP 测试的操作方式。

**undo operation** 命令用来恢复缺省情况。

### 【命令】

```
operation { get | put }
undo operation
```

### 【缺省情况】

FTP 测试的操作方式为 **get** 操作。

### 【视图】

FTP 测试类型视图

FTP 类型的模板视图

### 【缺省用户角色】

network-admin

### 【参数】

**get**: 从 FTP 服务器获取文件。

**put**: 向 FTP 服务器传送文件。

### 【使用指导】

进行 **put** 操作时，若配置了 **filename**，发送数据前判断 **filename** 指定的文件是否存在，如果存在则上传该文件，如果不存在则探测失败。

进行 **get** 操作时，如果 FTP 服务器上没有以 **url** 中所配置的文件名为名称的文件，则测试不会成功。进行 **get** 操作时，设备上不会保存从服务器获取的文件。

进行 **get**、**put** 操作时，请选用较小的文件进行测试，如果文件较大，可能会因为超时而导致测试失败，或由于占用较多的网络带宽而影响其他业务。

### 【举例】

# 在 FTP 测试类型下配置 FTP 测试的操作方式为 **put** 操作。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type ftp
[Sysname-nqa-admin-test-ftp] operation put
```

# 在 FTP 类型的 NQA 模板视图下配置测试的操作方式为 **put** 操作。

```
<Sysname> system-view
[Sysname] nqa template ftp ftptplt
[Sysname-nqatplt-ftp-ftptplt] operation put
```

### 【相关命令】

- **password**
- **username**

## 1.1.37 operation (HTTP/HTTPS test type view)

**operation** 命令用来配置 HTTP/HTTPS 测试的操作方式。

**undo operation** 命令用来恢复缺省情况。

### 【命令】

```
operation { get | post | raw }
undo operation
```

### 【缺省情况】

HTTP/HTTPS 测试的操作方式为 **get** 操作。

### 【视图】

HTTP 测试类型视图

HTTP/HTTPS 类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

**get**: 从 HTTP/HTTPS 服务器获取数据。  
**post**: 向 HTTP/HTTPS 服务器提交数据。  
**raw**: 向 HTTP/HTTPS 服务器发送 RAW 请求报文。

### 【使用指导】

HTTP/HTTPS 测试中以 HTTP/HTTPS 请求报文作为探测报文。

测试的操作方式为 **get** 或 **post** 时，请求报文内容从 url 中获取。

测试的操作方式为 **raw** 时，请求报文为 **raw-request** 子视图中配置的内容。在配置 HTTP/HTTPS 操作方式中的 **raw** 操作可以通过在 **raw-request** 子视图中配置 **get**、**post** 或其他服务器支持的请求类型的 HTTP/HTTPS 报文，以此直接作为探测报文。请确保 **raw-request** 子视图中所配置报文内容的正确性。

### 【举例】

# 在 HTTP 测试类型下配置 HTTP 测试的操作方式为 **raw** 操作。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type http
[Sysname-nqa-admin-test-http] operation raw
```

# 在 HTTP 类型的 NQA 模板视图下配置测试的操作方式为 **raw** 操作。

```
<Sysname> system-view
[Sysname] nqa template http httptplt
[Sysname-nqatplt-http-httptplt] operation raw
```

### 【相关命令】

- **password**
- **raw-request**
- **username**

## 1.1.38 out interface

**out interface** 命令用来指定探测报文的出接口。

**undo out interface** 命令用来恢复缺省情况。

### 【命令】

```
out interface interface-type interface-number
undo out interface
```

### 【缺省情况】

未指定探测报文的出接口。

### 【视图】

ICMP-echo 测试类型视图

DHCP 测试类型视图

UDP-tracert 测试类型视图

### 【缺省用户角色】

network-admin

### 【参数】

*interface-type interface-number*: 探测报文出接口的接口类型和接口编号。

### 【使用指导】

该命令指定的接口必须处于 UP 状态，否则 NQA 探测过程将会失败。

对于 ICMP-echo 测试类型，如果配置 **next-hop** 命令，此配置不生效。

### 【举例】

# 在 UDP-tracert 测试类型下配置 VLAN 接口 1 作为 UDP-tracert 探测报文的出接口。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type udp-tracert
[Sysname-nqa-admin-test-udp-tracert] out interface vlan-interface 1
```

## 1.1.39 password

**password** 命令用来配置测试服务器的用户登录密码。

**undo password** 命令用来恢复缺省情况。

### 【命令】

```
password { cipher | simple } string
undo password
```

### 【缺省情况】

未配置测试服务器的用户登录密码。

### 【视图】

FTP/HTTP 测试类型视图

FTP/HTTP/HTTPS/RADIUS 类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

**cipher**: 以密文方式设置密码。

**simple**: 以明文方式设置密码，该密码将以密文形式存储。

*string*: 密码字符串，区分大小写。对于 FTP/HTTP/HTTPS，明文密码为 1~32 个字符的字符串，密文密码为 1~73 个字符的字符串；对于 RADIUS，明文密码为 1~64 个字符的字符串，密文密码为 1~117 个字符的字符串。

### 【举例】

# 在 FTP 测试类型下配置 FTP 服务器的用户登录密码为 ftpuser。

```
<Sysname> system-view
[Sysname] nqa entry admin test
```



```
[Sysname-nqa-admin-test] type ftp
[Sysname-nqa-admin-test-ftp] password simple ftpuser
# 在 FTP 类型的 NQA 模板视图下配置 FTP 服务器的用户登录密码为 ftpuser。
<Sysname> system-view
[Sysname] nqa template ftp ftptplt
[Sysname-nqatplt-ftp-ftptplt] password simple ftpuser
```

#### 【相关命令】

- **operation**
- **username**

### 1.1.40 probe count

**probe count** 命令用来配置一次 NQA 测试中探测的次数。

**undo probe count** 命令用来恢复缺省情况。

#### 【命令】

```
probe count times
undo probe count
```

#### 【缺省情况】

对于 UDP-tracert 测试类型，对于一个 TTL 值的节点发送的探测报文次数为 3 次；其他类型的 NQA 测试一次 NQA 测试中的探测次数为 1 次。

#### 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图  
DHCP/DLSw/DNS/FTP/HTTP/SNMP 测试类型视图  
UDP-tracert 测试类型视图  
ICMP-jitter/UDP-jitter 测试类型视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**times**: 对于 UDP-tracert 测试类型，表示对于一个 TTL 值的节点发送的探测报文次数，取值范围为 1~10；对于其他测试类型，表示一次 NQA 测试中进行探测的次数，取值范围为 1~15。

#### 【使用指导】

不同测试类型中，探测的含义不同：

- 对于 TCP 和 DLSw 测试，一次探测操作是指建立一次连接。
- 对于 ICMP-jitter 和 UDP-jitter 测试，一次探测操作是指连续发送多个探测报文，发送探测报文的个数由 **probe packet-number** 命令指定。
- 对于 FTP、HTTP、DHCP 和 DNS 测试，一次探测操作是指完成一次相应的功能，例如上传或下载一个文件，获取一个 Web 页面，为接口申请一个 IP 地址，将一个域名解析为 IP 地址。
- 对于 ICMP-echo 和 UDP-echo 测试，一次探测操作是指发送一个探测报文。

- 对于 SNMP 测试，一次探测操作是指发送三个 SNMP 协议报文，分别对应 SNMP v1、SNMP v2c 和 SNMP v3 三个版本。
  - 对于 UDP-tracert 测试，一次探测操作是指一个特定 TTL 值的节点发送一个探测报文的操作。对于 UDP-tracert 测试，同一个 TTL 的节点，设备会发送数量为 **probe count** 命令配置的探测报文，系统在进行第一次探测之后，等待回应；对于其他类型的测试，如果配置的次数大于 1，那么系统在进行第一次探测之后，等待回应。如果到达 **probe timeout** 命令指定的探测超时时间时，仍然没有收到回应，则发起第二次探测。如此反复，直到完成指定次数的探测。
- Voice 和 Path-jitter 测试不支持该命令，一次测试中只能进行一次探测。

#### 【举例】

# 在 ICMP-echo 测试类型下配置一次 ICMP-echo 测试中探测的次数为 10 次。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] probe count 10
```

### 1.1.41 probe packet-interval

**probe packet-interval** 命令用来配置发送探测报文的时间间隔。

**undo probe packet-interval** 命令用来恢复缺省情况。

#### 【命令】

```
probe packet-interval interval
undo probe packet-interval
```

#### 【缺省情况】

发送探测报文的时间间隔为 20 毫秒。

#### 【视图】

ICMP-jitter/Path-jitter/UDP-jitter/Voice 测试类型视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**interval**: 测试中发送探测报文的时间间隔，取值范围为 10~60000，单位为毫秒。

#### 【举例】

# 在 UDP-jitter 测试类型下配置测试中发送探测报文的时间间隔为 100 毫秒。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type udp-jitter
[Sysname-nqa-admin-test-udp-jitter] probe packet-interval 100
```

### 1.1.42 probe packet-number

**probe packet-number** 命令用来配置一次探测中发送探测报文的个数。

**undo probe packet-number** 命令用恢复缺省情况。

### 【命令】

```
probe packet-number packet-number
undo probe packet-number
```

### 【缺省情况】

一次 ICMP-jitter、UDP-jitter 或 Path-jitter 探测中发送 10 个探测报文；一次 Voice 探测中发送 1000 个探测报文。

### 【视图】

ICMP-jitter/Path-jitter/UDP-jitter/Voice 测试类型视图

### 【缺省用户角色】

network-admin

### 【参数】

*packet-number*: 一次探测中发送探测报文的个数，对于 ICMP-jitter、UDP-jitter 和 Path-jitter 测试，取值范围为 10~1000；对于 Voice 测试，取值范围为 10~60000。

### 【举例】

# 在 UDP-jitter 测试类型下配置一次探测中发送 100 个探测报文。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type udp-jitter
[Sysname-nqa-admin-test-udp-jitter] probe packet-number 100
```

## 1.1.43 probe packet-timeout

**probe packet-timeout** 命令用来配置一次探测中等待响应报文的超时时间。

**undo probe packet-timeout** 命令用来恢复缺省情况。

### 【命令】

```
probe packet-timeout timeout
undo probe packet-timeout
```

### 【缺省情况】

ICMP-jitter、UDP-jitter 和 Path-jitter 测试中等待响应报文的超时时间为 3000 毫秒；Voice 测试中等待响应报文的超时时间为 5000 毫秒。

### 【视图】

ICMP-jitter/Path-jitter/UDP-jitter/Voice 测试类型视图

### 【缺省用户角色】

network-admin

### 【参数】

*timeout*: 一次探测中等待响应报文的超时时间，取值范围为 10~3600000，单位为毫秒。

### 【举例】

# 在 UDP-jitter 测试类型下配置一次探测中等待响应报文的超时时间为 100 毫秒。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type udp-jitter
[Sysname-nqa-admin-test-udp-jitter] probe packet-timeout 100
```

#### 1.1.44 probe timeout

**probe timeout** 命令用来配置探测的超时时间。

**undo probe timeout** 命令用来恢复缺省情况。

##### 【命令】

```
probe timeout timeout
undo probe timeout
```

##### 【缺省情况】

探测的超时时间为 3000 毫秒。

##### 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图  
DHCP/DLSw/DNS/FTP/HTTP/SNMP 测试类型视图  
UDP-tracert 测试类型视图  
任意类型的 NQA 模板视图

##### 【缺省用户角色】

network-admin

##### 【参数】

**timeout**: 一次探测的超时时间, 单位为毫秒。在 FTP、HTTP 探测中, 取值范围为 10~86400000; 在 DHCP、DLSw、DNS、ICMP-echo、SNMP、TCP、UDP-echo 和 UDP-tracert 探测中, 取值范围为 10~3600000。对于 NQA 模板类型来说, FTP、HTTP 和 HTTPS 模板的范围为 10~86400000, 其他模板范围为 10~3600000。

##### 【使用指导】

如果 NQA 探测没有在 **probe timeout** 命令指定的时间内完成, 则认为本次探测超时。

##### 【举例】

# 在 DHCP 测试类型下配置探测的超时时间为 10000 毫秒。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type dhcp
[Sysname-nqa-admin-test-dhcp] probe timeout 10000
```

# 在 HTTP 类型的 NQA 模板视图下配置探测的超时时间为 10000 毫秒。

```
<Sysname> system-view
[Sysname] nqa template http httptplt
[Sysname-nqatplt-http-httptplt] probe timeout 10000
```

### 1.1.45 raw-request

**raw-request** 命令用来进入 raw-request 子视图，并在该子视图下配置 HTTP/HTTPS 测试请求报文内容。

**undo raw-request** 命令用来恢复缺省情况。

#### 【命令】

```
raw-request
undo raw-request
```

#### 【缺省情况】

未配置 HTTP/HTTPS 测试请求报文的内容。

#### 【视图】

HTTP 测试类型视图  
HTTP/HTTPS 类型的 NQA 模板视图

#### 【缺省用户角色】

network-admin

#### 【使用指导】

每次使用 **raw-request** 命令进入 raw-request 子视图时，之前在该子视图下配置的 HTTP 测试请求报文内容会被清除。

HTTP/HTTPS 测试的操作方式为 **raw** 时，必须配置该命令。raw-request 子视图中所配置报文内容为 HTTP/HTTPS 的请求报文，需要确保报文的正确性，否则探测将失败。

#### 【举例】

# 进入 raw-request 子视图，并在该子视图下配置 HTTP 测试 GET 操作的请求报文的内容。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type http
[Sysname-nqa-admin-test-http] raw-request
[Sysname-nqa-admin-test-http-raw-request] GET /sdn/ui/app/index HTTP/1.0\r\nHost:
172.0.0.2\r\n\r\n
```

# 在 HTTP 类型的 NQA 模板视图下，进入 raw-request 子视图，并在该子视图下配置 HTTP 测试 POST 操作的请求报文的内容。

```
<Sysname> system-view
[Sysname] nqa template http httptplt
[Sysname-nqatplt-http-httptplt] raw-request
[Sysname-nqatplt-http-httptplt-raw-request] POST /sdn/ui/app/index HTTP/1.0\r\nHost:
172.0.0.2\r\nAuthorization: Basic cm9vdDoxMjM0NTY=\r\n\r\n
```

### 1.1.46 reaction checked-element { jitter-ds | jitter-sd }

**reaction checked-element { jitter-ds | jitter-sd }** 命令用来创建监测单向抖动时间的阈值告警组。

**undo reaction** 命令用来删除指定的阈值告警组。

## 【命令】

```
reaction item-number checked-element { jitter-ds | jitter-sd }
threshold-type { accumulate accumulate-occurrences | average }
threshold-value upper-threshold lower-threshold [ action-type { none |
trap-only } ]
undo reaction item-number
```

## 【缺省情况】

不存在监测单向抖动时间的阈值告警组。

## 【视图】

ICMP-jitter/UDP-jitter/Voice 测试类型视图

## 【缺省用户角色】

network-admin

## 【参数】

**item-number**: 阈值告警组编号，取值范围为 1~10。

**jitter-ds**: 监测从目的到源的单向抖动时间。

**jitter-sd**: 监测从源到目的的单向抖动时间。

**threshold-type**: 指定阈值类型。

**accumulate accumulate-occurrences**: 每次测试中，累计的单向抖动时间超出阈值的报文个数。对于 ICMP-jitter 与 UDP-jitter 测试，取值为 1~14999；对于 Voice 测试，取值范围为 1~59999。

**average**: 每次测试中，单向抖动时间的平均值。

**threshold-value**: 指定阈值范围。

**upper-threshold**: 阈值上限，取值范围为 0~3600000，单位为毫秒。

**lower-threshold**: 阈值下限，取值范围为 0~3600000，且必须小于等于阈值上限，单位为毫秒。

**action-type**: 触发的动作类型，缺省动作类型为 none。

**none**: 只在显示信息中记录监测结果，不向网管发送 Trap 消息。

**trap-only**: 条件满足时，在显示信息中记录监测结果的同时，向网管发送 Trap 消息。

## 【使用指导】

阈值告警组创建后，不能再通过 **reaction** 命令修改该阈值告警组的内容。若要修改阈值告警组的内容，则需要先通过 **undo reaction** 命令用来删除阈值告警组，再利用新的参数创建阈值告警组。

监测的对象是探测成功的报文，探测失败的报文不参与计数。

## 【举例】

# 在 UDP-jitter 测试类型下创建编号为 1 的阈值告警组，监测 UDP-jitter 探测报文的从目的到源的单向抖动时间，阈值上限为 50 毫秒，下限为 5 毫秒。NQA 测试组启动前，初始的阈值状态为 invalid。每次测试结束后，检查本次测试的平均单向抖动时间，若超出阈值，阈值状态置为 over-threshold；反之，置为 below-threshold。当阈值状态改变时，向网管发送 Trap 消息。

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

```
[Sysname] nqa entry admin test
```

```
[Sysname-nga-admin-test] type udp-jitter
[Sysname-nga-admin-test-udp-jitter] reaction 1 checked-element jitter-ds threshold-type
average threshold-value 50 5 action-type trap-only
# 在 UDP-jitter 测试类型下创建编号为 2 的阈值告警组，监测 UDP-jitter 探测报文的从目的到源的
单向抖动时间，阈值上限为 50 毫秒，下限为 5 毫秒。NQA 测试组启动前，初始的阈值状态为 invalid。
每次测试结束后，检查本次测试中累计的单向抖动时间超出阈值的报文个数，若达到或超过 100 个，
阈值状态置为 over-threshold；反之，置为 below-threshold。当阈值状态改变时，向网管发送 Trap
消息。
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nga-admin-test] type udp-jitter
[Sysname-nga-admin-test-udp-jitter] reaction 2 checked-element jitter-ds threshold-type
accumulate 100 threshold-value 50 5 action-type trap-only
```

### 1.1.47 reaction checked-element { owd-ds | owd-sd }

**reaction checked-element { owd-ds | owd-sd }** 命令用来创建监测单向时延的阈值告警组。

**undo reaction** 命令用来删除指定的阈值告警组。

#### 【命令】

```
reaction item-number checked-element { owd-ds | owd-sd } threshold-value
upper-threshold lower-threshold
undo reaction item-number
```

#### 【缺省情况】

不存在监测单向时延的阈值告警组。

#### 【视图】

ICMP-jitter/UDP-jitter/Voice 测试类型视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**item-number**: 阈值告警组编号，取值范围为 1~10。

**owd-ds**: 监测每个探测报文的从目的到源的单向时延。

**owd-sd**: 监测每个探测报文的从源到目的的单向时延。

**threshold-value**: 指定阈值范围。

**upper-threshold**: 阈值上限，取值范围为 0~3600000，单位为毫秒。

**lower-threshold**: 阈值下限，取值范围为 0~3600000，单位为毫秒，且必须小于等于阈值上限。

#### 【使用指导】

阈值告警组创建后，不能再通过 **reaction** 命令修改该阈值告警组的内容。若要修改阈值告警组的内容，则需要先通过 **undo reaction** 命令用来删除阈值告警组，再利用新的参数创建阈值告警组。

监测的对象是探测成功的报文，探测失败的报文不参与计数。

监测单向时延的阈值告警组不支持触发动作，但可以通过相关显示命令 **display nqa reaction counters** 和 **display nqa statistics** 显示当前的监测结果。

#### 【举例】

# 在 UDP-jitter 测试类型下创建编号为 1 的阈值告警组，监测每个 UDP-jitter 探测报文的从目的到源的单向时延，阈值上限为 50 毫秒，下限为 5 毫秒。NQA 测试组启动前，初始的阈值状态为 **invalid**。收到探测报文的应答报文后，计算该探测报文从目的到源的单向时延，若超出阈值范围，阈值状态置为 **over-threshold**；反之，置为 **below-threshold**。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type udp-jitter
[Sysname-nqa-admin-test-udp-jitter] reaction 1 checked-element owd-ds threshold-value 50 5
```

### 1.1.48 reaction checked-element icpif

**reaction checked-element icpif** 命令用来创建监测 Voice 测试 ICPIF 值的阈值告警组。

**undo reaction** 命令用来删除指定的阈值告警组。

#### 【命令】

```
reaction item-number checked-element icpif threshold-value upper-threshold
lower-threshold [ action-type { none | trap-only } ]
undo reaction item-number
```

#### 【缺省情况】

不存在监测 Voice 测试 ICPIF 值的阈值告警组。

#### 【视图】

Voice 测试类型视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**item-number**: 阈值告警组编号，取值范围为 1~10。

**threshold-value**: 指定阈值范围。

**upper-threshold**: 阈值上限，取值范围为 1~100。

**lower-threshold**: 阈值下限，取值范围为 1~100，且必须小于等于阈值上限。

**action-type**: 触发的动作类型，缺省动作类型为 **none**。

**none**: 只在显示信息中记录监测结果，不向网管发送 Trap 消息。

**trap-only**: 条件满足时，在显示信息中记录监测结果的同时，向网管发送 Trap 消息。

#### 【使用指导】

阈值告警组创建后，不能再通过 **reaction** 命令修改该阈值告警组的内容。若要修改阈值告警组的内容，则需要先通过 **undo reaction** 命令用来删除阈值告警组，再利用新的参数创建阈值告警组。



### 【举例】

# 在 Voice 测试类型下创建编号为 1 的阈值告警组，监测每次 Voice 测试的 ICPIF 值，阈值上限为 50，下限为 5。NQA 测试组启动前，初始的阈值状态为 *invalid*。每次测试结束后，检查本次测试的 ICPIF 值，若超出阈值范围，阈值状态置为 *over-threshold*；反之，置为 *below-threshold*。当阈值状态改变时，向网管发送 Trap 消息。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type voice
[Sysname-nqa-admin-test-voice] reaction 1 checked-element icpif threshold-value 50 5
action-type trap-only
```

### 1.1.49 reaction checked-element mos

**reaction checked-element mos** 命令用来创建监测 Voice 测试 MOS 值的阈值告警组。

**undo reaction** 命令用来删除指定的阈值告警组。

### 【命令】

```
reaction item-number checked-element mos threshold-value upper-threshold
lower-threshold [action-type { none | trap-only } ]
undo reaction item-number
```

### 【缺省情况】

不存在监测 Voice 测试 MOS 值的阈值告警组。

### 【视图】

Voice 测试类型视图

### 【缺省用户角色】

network-admin

### 【参数】

**item-number**: 阈值告警组的编号，取值范围为 1~10。

**threshold-value**: 指定阈值范围。

**upper-threshold**: 阈值上限，取值范围为 1~500。

**lower-threshold**: 阈值下限，取值范围为 1~500，且必须小于等于阈值上限。

**action-type**: 触发的动作类型，缺省动作类型为 **none**。

**none**: 只在显示信息中记录监测结果，不向网管发送 Trap 消息。

**trap-only**: 条件满足时，在显示信息中记录监测结果的同时，向网管发送 Trap 消息。

### 【使用指导】

实际的阈值下限（或阈值上限）为输入的阈值下限/100（或阈值上限/100），即如果输入的阈值下限和阈值上限分别为 100、200，则 MOS 值在 1~2 之间时，未超出阈值。

阈值告警组创建后，不能再通过 **reaction** 命令修改该阈值告警组的内容。若要修改阈值告警组的内容，则需要先通过 **undo reaction** 命令用来删除阈值告警组，再利用新的参数创建阈值告警组。

### 【举例】

# 在 Voice 测试类型下创建编号为 1 的阈值告警组，监测每次 Voice 测试的 MOS 值，阈值上限为 200，下限为 100。NQA 测试组启动前，初始的阈值状态为 invalid。每次测试结束后，检查本次测试的 MOS 值，若超出阈值范围，阈值状态置为 over-threshold；反之，置为 below-threshold。当阈值状态改变时，向网管发送 Trap 消息。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type voice
[Sysname-nqa-admin-test-voice] reaction 1 checked-element mos threshold-value 200 100
action-type trap-only
```

### 1.1.50 reaction checked-element packet-loss

**reaction checked-element packet-loss** 命令用来创建监测每次测试中丢包数的阈值告警组。

**undo reaction** 命令用来删除指定的阈值告警组。

### 【命令】

```
reaction item-number checked-element packet-loss threshold-type accumulate
accumulate-occurrences [ action-type { none | trap-only } ]
undo reaction item-number
```

### 【缺省情况】

不存在监测每次测试中丢包数的阈值告警组。

### 【视图】

ICMP-jitter/UDP-jitter/Voice 测试类型视图

### 【缺省用户角色】

network-admin

### 【参数】

**item-number**: 阈值告警组的编号，取值范围为 1~10。

**threshold-type**: 指定阈值类型。

**accumulate** **accumulate-occurrences**: 每次测试中，累计的丢包数。对于 ICMP-jitter 与 UDP-jitter 测试，取值范围为 1~15000；对于 Voice 测试，取值范围为 1~60000。

**action-type**: 触发的动作类型，缺省动作类型为 **none**。

**none**: 只在显示信息中记录监测结果，不向网管发送 Trap 消息。

**trap-only**: 条件满足时，在显示信息中记录监测结果的同时，向网管发送 Trap 消息。

### 【使用指导】

阈值告警组创建后，不能再通过 **reaction** 命令修改该阈值告警组的内容。若要修改阈值告警组的内容，则需要先通过 **undo reaction** 命令用来删除阈值告警组，再利用新的参数创建阈值告警组。

### 【举例】

# 在 UDP-jitter 测试类型下创建编号为 1 的阈值告警组，监测每次 UDP-jitter 测试的丢包数。NQA 测试组启动前，初始的阈值状态为 **invalid**。每次测试结束后，检查本次测试中累计的丢包数，若达到或超过 100 个，阈值状态置为 **over-threshold**；反之，置为 **below-threshold**。当阈值状态改变时，向网管发送 Trap 消息。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type udp-jitter
[Sysname-nqa-admin-test-udp-jitter] reaction 1 checked-element packet-loss threshold-type
accumulate 100 action-type trap-only
```

### 1.1.51 reaction checked-element probe-duration

**reaction checked-element probe-duration** 命令用来创建监测探测持续时间的阈值告警组。

**undo reaction** 命令用来删除指定的阈值告警组。

### 【命令】

```
reaction item-number checked-element probe-duration threshold-type
{ accumulate accumulate-occurrences | average | consecutive
consecutive-occurrences } threshold-value upper-threshold lower-threshold
[ action-type { none | trap-only } ]
undo reaction item-number
```

### 【缺省情况】

不存在监测探测持续时间的阈值告警组。

### 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图

DHCP/DLSw/DNS/FTP/HTTP/SNMP 测试类型视图

### 【缺省用户角色】

network-admin

### 【参数】

**item-number**: 阈值告警组的编号，取值范围为 1～10。

**threshold-type**: 指定阈值类型。

**accumulate accumulate-occurrences**: 每次测试中，累计的探测持续时间超出阈值的探测次数。**accumulate-occurrences** 取值范围为 1～15。

**average**: 每次测试中，探测持续时间的平均值。

**consecutive consecutive-occurrences**: 测试组启动后，连续的探测持续时间超出阈值的探测次数。**consecutive-occurrences** 取值范围为 1～16。

**threshold-value**: 指定阈值范围。

**upper-threshold**: 阈值上限，取值范围为 0～3600000，单位为毫秒。

**lower-threshold:** 阈值下限，取值范围为 0~3600000，单位为毫秒，且必须小于等于阈值上限。

**action-type:** 触发的动作类型，缺省动作类型为 **none**。

**none:** 只在显示信息中记录监测结果，不向网管发送 Trap 消息。

**trap-only:** 条件满足时，在显示信息中记录监测结果的同时，向网管发送 Trap 消息。DNS 测试不支持发送 Trap，DNS 测试类型视图下无此参数。

### 【使用指导】

阈值告警组创建后，不能再通过 **reaction** 命令修改该阈值告警组的内容。若要修改阈值告警组的内容，则需要先通过 **undo reaction** 命令用来删除阈值告警组，再利用新的参数创建阈值告警组。

监测的对象是成功的探测，失败的探测不参与计数。

### 【举例】

# 在 ICMP-echo 测试类型下创建编号为 1 的阈值告警组，监测 ICMP-echo 探测的持续时间，阈值上限为 50 毫秒，下限为 5 毫秒。NQA 测试组启动前，初始的阈值状态为 **invalid**。每次测试结束后，检查本次测试的平均探测持续时间，若超出阈值，阈值状态置为 **over-threshold**；反之，置为 **below-threshold**。当阈值状态改变时，向网管发送 Trap 消息。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] reaction 1 checked-element probe-duration
threshold-type average threshold-value 50 5 action-type trap-only
```

# 在 ICMP-echo 测试类型下创建编号为 2 的阈值告警组，监测 ICMP-echo 探测的持续时间，阈值上限为 50 毫秒，下限为 5 毫秒。NQA 测试组启动前，初始的阈值状态为 **invalid**。每次测试结束后，检查本次测试中累计的持续时间超出阈值的探测次数，若达到或超过 10 次，阈值状态置为 **over-threshold**；反之，置为 **below-threshold**。当阈值状态改变时，向网管发送 Trap 消息。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] reaction 2 checked-element probe-duration
threshold-type accumulate 10 threshold-value 50 5 action-type trap-only
```

# 在 ICMP-echo 测试类型下创建编号为 3 的阈值告警组，监测 ICMP-echo 探测的持续时间，阈值上限为 50 毫秒，下限为 5 毫秒。NQA 测试组启动前，初始的阈值状态为 **invalid**。每次探测结束后，检查测试组启动以来连续的持续时间超出阈值的探测次数，若达到或超过 10 次，阈值状态置为 **over-threshold**；反之，置为 **below-threshold**。当阈值状态改变时，向网管发送 Trap 消息。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] reaction 3 checked-element probe-duration
threshold-type consecutive 10 threshold-value 50 5 action-type trap-only
```

## 1.1.52 reaction checked-element probe-fail (for trap)

**reaction checked-element probe-fail** 命令用来创建监测探测失败次数的阈值告警组。

**undo reaction** 命令用来删除指定的阈值告警组。

## 【命令】

```
reaction item-number checked-element probe-fail threshold-type { accumulate  
accumulate-occurrences | consecutive consecutive-occurrences }  
[ action-type { none | trap-only } ]  
undo reaction item-number
```

## 【缺省情况】

不存在监测探测失败次数的阈值告警组。

## 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图

DHCP/DLSw/DNS/FTP/HTTP/SNMP 测试类型视图

## 【缺省用户角色】

network-admin

## 【参数】

**item-number**: 阈值告警组编号，取值范围为 1~10。

**threshold-type**: 指定阈值类型。

**accumulate accumulate-occurrences**: 一次测试中，累计的探测失败次数。  
**accumulate-occurrences** 取值范围为 1~15。

**consecutive consecutive-occurrences**: NQA 测试组启动以来，连续的探测失败次数。  
**consecutive-occurrences** 取值范围为 1~16。

**action-type**: 触发的动作类型，缺省动作类型为 **none**。

**none**: 只在显示信息中记录监测结果，不向网管发送 Trap 消息。

**trap-only**: 条件满足时，在显示信息中记录监测结果的同时，向网管发送 Trap 消息。DNS 测试不支持发送 trap，DNS 测试类型视图下无此参数。

## 【使用指导】

阈值告警组创建后，不能再通过 **reaction** 命令修改该阈值告警组的内容。若要修改阈值告警组的内容，则需要先通过 **undo reaction** 命令用来删除阈值告警组，再利用新的参数创建阈值告警组。

## 【举例】

# 在 ICMP-echo 测试类型下创建编号为 1 的阈值告警组，监测 ICMP-echo 探测的失败次数。NQA 测试组启动前，初始的阈值状态为 **invalid**。每次测试结束后，检查本次测试中累计的探测失败次数，若达到或超过 10 次，阈值状态置为 **over-threshold**；反之，置为 **below-threshold**。当阈值状态改变时，向网管发送 Trap 消息。

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

```
[Sysname] nqa entry admin test
```

```
[Sysname-nqa-admin-test] type icmp-echo
```

```
[Sysname-nqa-admin-test-icmp-echo] reaction 1 checked-element probe-fail threshold-type  
accumulate 10 action-type trap-only
```

# 在 ICMP-echo 测试类型下创建编号为 2 的阈值告警组，监测 ICMP-echo 探测的失败次数。NQA 测试组启动前，初始的阈值状态为 **invalid**。每次探测结束后，检查测试组启动以来连续的探测失败

次数，若达到或超过 10 次，阈值状态置为 **over-threshold**；反之，置为 **below-threshold**。当阈值状态改变时，向网管发送 **Trap** 消息。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] reaction 2 checked-element probe-fail threshold-type
consecutive 10 action-type trap-only
```

### 1.1.53 reaction checked-element probe-fail (for trigger)

**reaction checked-element probe-fail** 命令用来建立联动项，对当前所在测试组中的探测进行监测，当连续探测失败次数达到阈值时，就触发其他模块联动。

**undo reaction** 命令用来删除指定的联动项。

#### 【命令】

```
reaction item-number checked-element probe-fail threshold-type consecutive
consecutive-occurrences action-type trigger-only
undo reaction item-number
```

#### 【缺省情况】

不存在联动项。

#### 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图

DHCP/DLSw/DNS/FTP/HTTP/SNMP 测试类型视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**item-number**: 联动项序号，取值范围为 1~10。

**threshold-type**: 指定门限类型。

**consecutive consecutive-occurrences**: NQA 测试组启动以来连续的探测失败次数，取值范围为 1~16。

**action-type**: 触发的动作类型。

**trigger-only**: 条件满足时，触发其它模块联动。

#### 【使用指导】

联动项创建后，不能再通过 **reaction** 命令修改该联动项的内容。若要修改联动项的内容，则需要先通过 **undo reaction** 命令用来删除联动项，再利用新的参数创建联动项。

#### 【举例】

# 在 TCP 测试类型下建立序号为 1 的联动项，连续探测失败 3 次，触发其他模块联动。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type tcp
```

```
[Sysname-nqa-admin-test-tcp] reaction 1 checked-element probe-fail threshold-type  
consecutive 3 action-type trigger-only
```

#### 【相关命令】

- **track**（可靠性命令参考/Track）

### 1.1.54 reaction checked-element rtt

**reaction checked-element rtt** 命令用来创建监测报文往返时延的阈值告警组。

**undo reaction** 命令用来删除指定的阈值告警组。

#### 【命令】

```
reaction item-number checked-element rtt threshold-type { accumulate  
accumulate-occurrences | average } threshold-value upper-threshold  
lower-threshold [ action-type { none | trap-only } ]  
undo reaction item-number
```

#### 【缺省情况】

不存在监测报文往返时延的阈值告警组。

#### 【视图】

ICMP-jitter/UDP-jitter/Voice 测试类型视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**item-number**: 阈值告警组编号，取值范围为 1~10。

**threshold-type**: 指定阈值类型。

**accumulate accumulate-occurrences**: 每次测试中，累计的 RTT 超出阈值的报文个数。对于 ICMP-jitter 与 UDP-jitter 测试，取值范围为 1~15000；对于 Voice 测试，取值范围为 1~60000。

**average**: 每次测试中，报文往返时间的平均值。

**threshold-value**: 指定阈值范围。

**upper-threshold**: 阈值上限，取值范围为 0~3600000，单位为毫秒。

**lower-threshold**: 阈值下限，取值范围为 0~3600000，单位为毫秒，且必须小于等于阈值上限。

**action-type**: 触发的动作类型，缺省动作类型为 **none**。

**none**: 只在显示信息中记录监测结果，不向网管发送 Trap 消息。

**trap-only**: 条件满足时，在显示信息中记录监测结果的同时，向网管发送 Trap 消息。

#### 【使用指导】

阈值告警组创建后，不能再通过 **reaction** 命令修改该阈值告警组的内容。若要修改阈值告警组的内容，则需要先通过 **undo reaction** 命令用来删除阈值告警组，再利用新的参数创建阈值告警组。

监测的对象是探测成功的报文，探测失败的报文不参与计数。



### 【举例】

# 在 UDP-jitter 测试类型下创建编号为 1 的阈值告警组, 监测 UDP-jitter 探测报文的往返时间, 阈值上限为 50 毫秒, 下限为 5 毫秒。NQA 测试组启动前, 初始的阈值状态为 *invalid*。每次测试结束后, 检查本次测试的平均报文往返时间, 若超出阈值, 阈值状态置为 *over-threshold*; 反之, 置为 *below-threshold*。当阈值状态改变时, 向网管发送 Trap 消息。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type udp-jitter
[Sysname-nqa-admin-test-udp-jitter] reaction 1 checked-element rtt threshold-type average
threshold-value 50 5 action-type trap-only
```

# 在 UDP-jitter 测试类型下创建编号为 2 的阈值告警组, 监测每个 UDP-jitter 探测报文的往返时间, 阈值上限为 50 毫秒, 下限为 5 毫秒。NQA 测试组启动前, 初始的阈值状态为 *invalid*。每次测试结束后, 检查本次测试中累计的 RTT 超出阈值的报文个数, 若达到或超过 100 个, 阈值状态置为 *over-threshold*; 反之, 置为 *below-threshold*。当阈值状态改变时, 向网管发送 Trap 消息。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type udp-jitter
[Sysname-nqa-admin-test-udp-jitter] reaction 1 checked-element rtt threshold-type
accumulate 100 threshold-value 50 5 action-type trap-only
```

## 1.1.55 reaction trap

**reaction trap** 命令用来配置在指定条件下向网管服务器发送 Trap 消息。

**undo reaction trap** 命令用来恢复缺省情况。

### 【命令】

```
reaction trap { path-change | probe-failure consecutive-probe-failures |
test-complete | test-failure [ accumulate-probe-failures ] }
undo reaction trap { path-change | probe-failure | test-complete |
test-failure }
```

### 【缺省情况】

不向网管服务器发送 Trap 消息。

### 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图

DHCP/DLSw/DNS/FTP/HTTP/SNMP 测试类型视图

UDP-tracert 测试类型视图

ICMP-jitter/UDP-jitter/Voice 测试类型视图

### 【缺省用户角色】

network-admin

### 【参数】

**path-change**: 当进行 UDP-tracert 类型测试时, 在配置了 **frequency** 命令后进行连续测试后, 如果检测到当前路径相对于上一次测试路径发生变化, 则设备发送一次 Trap 消息。



**probe-failure consecutive-probe-failures:** 每次探测结束后，计算本次 NQA 测试中探测连续失败的次数，如果连续失败次数大于或等于 *consecutive-probe-failures*，则向网管服务器发送探测失败的 Trap 消息。一次测试中，可能发送多次 Trap 消息。*consecutive-probe-failures* 为一次测试中连续探测失败的次数，取值范围为 1~15。

**test-complete:** 对于非 UDP-tracert 类型测试，当测试完成时发送测试完成的 Trap 消息。对于 UDP-tracert 类型测试，测试出到达目的设备的路径后，发送测试完成的 Trap 消息。

**test-failure accumulate-probe-failures:** 对于非 UDP-tracert 类型测试，一次 NQA 测试结束后，计算本次 NQA 测试中探测失败的累计次数，如果累计失败次数大于或等于 *accumulate-probe-failures*，则向网管服务器发送测试失败的 Trap 消息。*accumulate-probe-failures* 为一次测试中累计探测失败的次数，为必须输入的参数，取值范围为 1~15。对于 UDP-tracert 类型测试，只要未能测试出到达目的地的路径，就发送一次 Trap 消息。用户不能输入参数 *accumulate-probe-failures*。

#### 【使用指导】

ICMP-jitter、UDP-jitter 和 Voice 测试只支持 **reaction trap test-complete**。

UDP-tracert 测试支持 **reaction trap path-change**，**reaction trap test-complete** 和 **reaction trap test-failure**（其中 UDP-tracert 测试不支持 **reaction trap test-failure** 的 *accumulate-probe-failures* 参数）。

#### 【举例】

# 在 ICMP-echo 测试类型下配置测试中连续探测失败次数大于或等于 5 次时，发送探测失败的 Trap 消息。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] reaction trap probe-failure 5
```

### 1.1.56 reaction trigger per-probe

**reaction trigger per-probe** 命令用来配置每次探测结果发送机制，即每次探测结束时都会将探测结果发送给外部特性，使外部特性利用 NQA 测试的结果进行相应处理。

**undo reaction trigger per-probe** 命令用来恢复缺省情况。

#### 【命令】

```
reaction trigger per-probe
undo reaction trigger per-probe
```

#### 【缺省情况】

连续探测成功或失败 3 次时，NQA 客户端会把探测成功或失败的消息发送给外部特性，使外部特性利用 NQA 测试的结果进行相应处理。

#### 【视图】

ICMP/TCP Half Open 类型的 NQA 模板视图

#### 【缺省用户角色】

network-admin

### 【使用指导】

本命令与 **reaction trigger probe-fail** 命令作用相同，多次执行这两条命令时，最后一次执行的命令生效。

本命令与 **reaction trigger probe-pass** 命令作用相同，多次执行这两条命令时，最后一次执行的命令生效。

### 【举例】

# 在 ICMP 类型的 NQA 模板视图下配置每次探测结果发送机制，即每次探测结束时都会将探测结果发送给外部特性，使外部特性利用 NQA 测试的结果进行相应处理。

```
<Sysname> system-view
[Sysname] nqa template icmp icmptplt
[Sysname-nqatplt-icmp-icmptplt] reaction trigger per-probe
```

### 【相关命令】

- **reaction trigger probe-fail**
- **reaction trigger probe-pass**

## 1.1.57 reaction trigger probe-fail

**reaction trigger probe-fail** 命令用来配置连续探测失败的次数。

**undo reaction trigger probe-fail** 命令用来恢复缺省情况。

### 【命令】

```
reaction trigger probe-fail count
undo reaction trigger probe-fail
```

### 【缺省情况】

连续探测失败 3 次时，NQA 客户端会把探测失败的消息发送给外部特性，使外部特性利用 NQA 测试的结果进行相应处理。

### 【视图】

任意类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

*count*：连续探测失败的次数，取值范围为 1~15。

### 【使用指导】

当连续探测失败次数达到命令配置的数值时，NQA 客户端会把探测失败的消息发送给外部特性，使外部特性利用 NQA 测试的结果进行相应处理。

外部特性调用 NQA 模板后进行相应的 NQA 测试，使用此命令可以设定节点失效的连续测试失败次数。

本命令与 **reaction trigger per-probe** 命令作用相同，多次执行这两条命令时，最后一次执行的命令生效。

### 【举例】

# 在 HTTP 类型的 NQA 模板视图下配置确定节点失效需要连续探测失败 5 次。当连续探测失败的次数达到 5 次时，NQA 客户端把探测失败的消息发送给外部特性，使外部特性利用 NQA 测试的结果进行相应处理。

```
<Sysname> system-view
[Sysname] nqa template http httptplt
[Sysname-nqatplt-http-httptplt] reaction trigger probe-fail 5
```

### 【相关命令】

- **reaction trigger per-probe**
- **reaction trigger probe-pass**

## 1.1.58 reaction trigger probe-pass

**reaction trigger probe-pass** 命令用来配置连续探测成功次数。

**undo reaction trigger probe-pass** 命令用来恢复缺省情况。

### 【命令】

```
reaction trigger probe-pass count
undo reaction trigger probe-pass
```

### 【缺省情况】

连续探测成功 3 次时，NQA 客户端会把探测成功的消息发送给外部特性，使外部特性利用 NQA 测试的结果进行相应处理。

### 【视图】

任意类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

**count**: 连续探测成功的次数，取值范围为 1~15。

### 【使用指导】

当连续探测成功次数达到命令配置的数值时，NQA 客户端会把探测成功的消息发送给外部特性，使外部特性利用 NQA 测试的结果进行相应处理。

外部特性调用 NQA 模板后进行相应的 NQA 测试，使用此命令可以设定节点有效的连续探测成功次数。

本命令与 **reaction trigger per-probe** 命令作用相同，多次执行这两条命令时，最后一次执行的命令生效。

### 【举例】

# 在 HTTP 类型的 NQA 模板视图下配置确定节点有效需要连续探测成功 5 次。当连续探测成功的次数达到 5 次时，NQA 客户端把探测成功的消息发送给外部特性，使外部特性利用 NQA 测试的结果进行相应处理。

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

```
[Sysname] nqa template http httptplt
[Sysname-nqatplt-http-httptplt] reaction trigger probe-pass 5
```

#### 【相关命令】

- **reaction trigger per-probe**
- **reaction trigger probe-fail**

### 1.1.59 resolve-target

**resolve-target** 命令用来配置要解析的域名。

**undo resolve-target** 命令用来恢复缺省情况。

#### 【命令】

```
resolve-target domain-name
undo resolve-target
```

#### 【缺省情况】

未配置要解析的域名。

#### 【视图】

DNS 测试类型视图

DNS 类型的 NQA 模板视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*domain-name*: 要解析的域名，由“.”分隔的字符串组成（如 aabbcc.com），每个字符串的长度不超过 63 个字符，包括“.”在内的总长度不超过 255 个字符，区分大小写。字符串中可以包含字母、数字、“-”及“\_”，不能出现连续“.”。

#### 【举例】

# 在 DNS 测试类型下配置 DNS 测试要解析的域名为 domain1。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type dns
[Sysname-nqa-admin-test-dns] resolve-target domain1
```

# 在 DNS 类型的 NQA 模板视图下配置测试要解析的域名为 domain1。

```
<Sysname> system-view
[Sysname] nqa template dns dnstplt
[Sysname-nqatplt-dns-dnstplt] resolve-target domain1
```

### 1.1.60 resolve-type

**resolve-type** 命令用来配置域名解析类型。

**undo resolve-type** 命令用来恢复缺省情况。

### 【命令】

```
resolve-type { A | AAAA }  
undo resolve-type
```

### 【缺省情况】

域名解析类型为 **A** 类型。

### 【视图】

DNS 类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

**A**: 域名解析类型为 A 类型请求，即将域名解析为 IPv4 地址。

**AAAA**: 域名解析类型为 AAAA 类型请求，即将域名解析为 IPv6 地址。

### 【举例】

# 在 DNS 类型的 NQA 模板视图下，配置测试的域名解析类型为 **A**。

```
<Sysname> system-view  
[Sysname] nqa template dns dnstplt  
[Sysname-nqatplt-dns-dnstplt] resolve-type A
```

## 1.1.61 route-option bypass-route

**route-option bypass-route** 命令用来开启路由表旁路功能，探测直连目的地的连通情况。

**undo route-option bypass-route** 命令用来关闭路由表旁路功能。

### 【命令】

```
route-option bypass-route  
undo route-option bypass-route
```

### 【缺省情况】

路由表旁路功能处于关闭状态。

### 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图  
DLSw/DNS/FTP/HTTP/SNMP 测试类型视图  
UDP-tracert 测试类型视图  
ICMP-jitter/UDP-jitter/Voice 测试类型视图

### 【缺省用户角色】

network-admin

### 【使用指导】

开启该功能后，将不进行路由查找，而直接将报文发送到直连网络的目的地。

开启该功能后，设备转发探测报文可以经过的最大跳数为 **1**，**ttl** 命令设置的跳数不会生效。

测试目的端使用 IPv6 地址时，本命令配置无效。

#### 【举例】

# 在 ICMP-echo 测试类型下开启路由旁路功能。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] route-option bypass-route
```

### 1.1.62 source interface (ICMP-echo/UDP-tracert test type view)

**source interface** 命令用来配置指定接口的 IP 地址作为探测报文的源 IP 地址。

**undo source interface** 命令用来恢复缺省情况。

#### 【命令】

```
source interface interface-type interface-number
undo source interface
```

#### 【缺省情况】

以报文发送接口的主 IP 地址作为探测报文中的源 IP 地址。

#### 【视图】

ICMP-echo 测试类型视图

UDP-tracert 测试类型视图

ICMP 类型的 NQA 模板视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*interface-type interface-number*: 探测报文源接口的接口类型和接口编号。

#### 【使用指导】

本命令指定的接口必须处于 UP 状态，否则探测将会失败。

对于 ICMP-echo 测试类型及 ICMP 类型的 NQA 模板，本命令与 **source ip**、**source ipv6** 命令作用相同，多次执行这三条命令时，最后一次执行的命令生效；对于 UDP-tracert 测试类型，本命令与 **source ip** 命令作用相同，多次执行这两条命令时，最后一次执行的命令生效。

#### 【举例】

# 在 ICMP-echo 测试类型下指定 VLAN 接口 1 的 IP 地址作为 ICMP-echo 探测报文的源 IP 地址。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] source interface vlan-interface 1
```

# 在 ICMP 类型的 NQA 模板视图下指定 VLAN 接口 1 的 IP 地址作为 ICMP-echo 探测报文的源 IP 地址。

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

```
[Sysname] nqa template icmp icmptplt  
[Sysname-nqatplt-icmp-icmptplt] source interface vlan-interface 1
```

#### 【相关命令】

- **source ip**
- **source ipv6**

### 1.1.63 source ip

**source ip** 命令用来配置探测报文的源 IPv4 地址。

**undo source ip** 命令用来恢复缺省情况。

#### 【命令】

```
source ip ip-address  
undo source ip
```

#### 【缺省情况】

以报文发送接口的主 IP 地址作为探测报文中的源 IPv4 地址。

#### 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图  
DHCP/DLSw/FTP/HTTP/SNMP 测试类型视图  
UDP-tracert 测试类型视图  
ICMP-jitter/Path-jitter/UDP-jitter/Voice 测试类型视图  
任意类型的 NQA 模板视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**ip-address**: 探测报文的源 IPv4 地址。

#### 【使用指导】

**source ip** 命令配置的源 IP 地址必须是设备上接口的 IPv4 地址，且接口为 UP 状态，否则测试将会失败。

对于 NQA 模板类型来说，当源地址类型和目的地址类型不一致时，以目的地址类型为准，进行该类型的报文探测，此时源地址的配置不生效。

对于 ICMP-echo/UDP-tracert 测试类型及 ICMP 类型的 NQA 模板，本命令和 **source interface** 命令作用相同，多次执行这两条命令时，最后一次执行的命令生效。

#### 【举例】

# 在 ICMP-echo 测试类型下配置 ICMP-echo 探测报文中的源 IPv4 地址为 10.1.1.1。

```
<Sysname> system-view  
[Sysname] nqa entry admin test  
[Sysname-nqa-admin-test] type icmp-echo  
[Sysname-nqa-admin-test-icmp-echo] source ip 10.1.1.1
```

# 在 ICMP 类型的 NQA 模板视图下配置探测报文中的源 IPv4 地址为 10.1.1.1。

```
<Sysname> system-view
[Sysname] nqa template icmp icmptplt
[Sysname-nqatplt-icmp-icmptplt] source ip 10.1.1.1
```

#### 【相关命令】

- **source interface**

### 1.1.64 source ipv6

**source ipv6** 命令用来配置探测报文的源 IPv6 地址。

**undo source ipv6** 命令用来恢复缺省情况。

#### 【命令】

```
source ipv6 ipv6-address
undo source ipv6
```

#### 【缺省情况】

以报文发送接口的 IPv6 地址作为探测报文中的源 IPv6 地址。

#### 【视图】

ICMP-echo 测试类型视图  
任意类型的 NQA 模板视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*ipv6-address*: 探测报文的源 IPv6 地址，不支持 IPv6 链路本地地址。

#### 【使用指导】

**source ipv6** 命令配置的源 IPv6 地址必须是设备上接口的 IPv6 地址，且接口为 up 状态，否则测试将会失败。

对于 NQA 模板类型来说，当源地址类型和目的地址类型不一致时，以目的地址类型为准，进行该类型的报文探测，此时源地址的配置不生效。

对于 ICMP-echo 测试类型及 ICMP 类型的 NQA 模板，本命令和 **source interface** 命令作用相同，多次执行这两条命令时，最后一次执行的命令生效。

#### 【举例】

# 在 ICMP-echo 测试类型下配置 ICMP-echo 探测报文中的源 IPv6 地址为 1::1。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] source ipv6 1::1
```

# 在 ICMP 类型的 NQA 模板视图下配置探测报文中的源 IPv6 地址为 1::1。

```
<Sysname> system-view
[Sysname] nqa template icmp icmptplt
[Sysname-nqatplt-icmp-icmptplt] source ipv6 1::1
```



## 【相关命令】

- `source interface`

### 1.1.65 source port

`source port` 命令用来配置探测报文的源端口号。

`undo source port` 命令用来恢复缺省情况。

## 【命令】

`source port port-number`

`undo source port`

## 【缺省情况】

未指定源端口号。

## 【视图】

UDP-echo 测试类型视图

SNMP 测试类型视图

UDP-tracert 测试类型视图

UDP-jitter/Voice 测试类型视图

DNS 类型的 NQA 模板视图

## 【缺省用户角色】

network-admin

## 【参数】

*port-number*: 探测报文的源端口号，取值范围为 1~65535。

## 【举例】

# 在 UDP-echo 测试类型下配置探测报文的源端口号为 8000。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type udp-echo
[Sysname-nqa-admin-test-udp-echo] source port 8000
```

# 在 DNS 类型的 NQA 模板视图下配置探测报文的源端口号为 8000。

```
<Sysname> system-view
[Sysname] nqa template dns dnstplt
[Sysname-nqatplt-dns-dnstplt] source port 8000
```

### 1.1.66 ssl-client-policy

`ssl-client-policy` 命令用来为 HTTPS/SSL 类型的 NQA 模板绑定 SSL 客户端策略。

`undo ssl-client-policy` 命令用来恢复缺省情况。

## 【命令】

`ssl-client-policy policy-name`

`undo ssl-client-policy`

### 【缺省情况】

未绑定 SSL 客户端策略。

### 【视图】

HTTPS/SSL 类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

*policy-name*: SSL 客户端策略名，为 1~31 个字符的字符串，不区分大小写。

### 【使用指导】

通过绑定 SSL 客户端策略与服务器建立 SSL 连接，确认服务器业务的可用性。

### 【举例】

# 在 SSL 类型的 NQA 模板视图下配置 SSL 客户端策略为 policy。

```
<Sysname> system-view
[Sysname] nqa template ssl ssltplt
[Sysname-nqatplt-ssl-ssltplt] ssl-client-policy policy
```

## 1.1.67 statistics hold-time

**statistics hold-time** 命令用来配置统计组的保留时间。

**undo statistics hold-time** 命令用来恢复缺省情况。

### 【命令】

```
statistics hold-time hold-time
undo statistics hold-time
```

### 【缺省情况】

统计组的保留时间为 120 分钟。

### 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图

DHCP/DLSw/DNS/FTP/HTTP/SNMP 测试类型视图

ICMP-jitter/Path-jitter/UDP-jitter/Voice 测试类型视图

### 【缺省用户角色】

network-admin

### 【参数】

*hold-time*: 统计组的保留时间，取值范围为 1~1440，单位为分钟。

### 【使用指导】

统计组具有老化功能。统计组保存一定时间后将被删除，以便记录新的统计组信息。

### 【举例】

# 在 ICMP-echo 测试类型下配置统计组的保留时间为 3 分钟。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] statistics hold-time 3
```

### 1.1.68 statistics interval

**statistics interval** 命令用来配置对测试结果进行统计的时间间隔。

**undo statistics interval** 命令用来恢复缺省情况。

#### 【命令】

```
statistics interval interval
undo statistics interval
```

#### 【缺省情况】

对测试结果进行统计的时间间隔为 60 分钟。

#### 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图  
DHCP/DLSw/DNS/FTP/HTTP/SNMP 测试类型视图  
ICMP-jitter/Path-jitter/UDP-jitter/Voice 测试类型视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**interval**: 对测试结果进行统计的时间间隔，取值范围为 1～35791394，单位为分钟。

#### 【使用指导】

NQA 将统计时间间隔内完成的 NQA 测试归为一组，计算该组测试结果的统计值，这些统计值构成一个统计组。通过 **display nqa statistics** 命令可以显示该统计组的信息。

#### 【举例】

# 在 ICMP-echo 测试类型下配置对测试结果进行统计的时间间隔为 2 分钟。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] statistics interval 2
```

### 1.1.69 statistics max-group

**statistics max-group** 命令用来配置能够保留的最大统计组个数。

**undo statistics max-group** 命令用来恢复缺省情况。

#### 【命令】

```
statistics max-group number
undo statistics max-group
```

### 【缺省情况】

能够保留的最大统计组个数为 2。

### 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图

DHCP/DLSw/DNS/FTP/HTTP/SNMP 测试类型视图

ICMP-jitter/Path-jitter/UDP-jitter/Voice 测试类型视图

### 【缺省用户角色】

network-admin

### 【参数】

**number**: 能够保留的最大统计组个数，取值范围为 0~100。

### 【使用指导】

当保留的统计组数目达到最大值时，如果形成新的统计组，保存时间最久的统计组将被删除。

能够保留的最大统计组个数为 0 时，不进行统计。

### 【举例】

# 在 ICMP-echo 测试类型下配置能够保留的最大统计组个数为 5。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] statistics max-group 5
```

## 1.1.70 target-only

**target-only** 命令用来配置 Path-jitter 测试中仅针对到达目的地址的完整路径进行探测，不逐跳进行探测。

**undo target-only** 命令用来恢复缺省情况。

### 【命令】

```
target-only
undo target-only
```

### 【缺省情况】

Path-jitter 测试中会逐跳进行探测。

### 【视图】

Path-jitter 测试类型视图

### 【缺省用户角色】

network-admin

### 【举例】

# 在 Path-jitter 测试类型下配置仅针对到达目的地址的完整路径进行探测。

```
<Sysname> system-view
[Sysname] nqa entry admin test
```

```
[Sysname-nqa-admin-test] type path-jitter
[Sysname-nqa-admin-test-path-jitter] target-only
```

### 1.1.71 tos

**tos** 命令用来配置 NQA 探测报文 IP 报文头中服务类型域的值。

**undo tos** 命令用来恢复缺省情况。

#### 【命令】

```
tos value
undo tos
```

#### 【缺省情况】

NQA 探测报文 IP 报文头中服务类型域的值 0。

#### 【视图】

任意测试类型视图

任意类型的 NQA 模板视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**value**: 探测报文 IP 报文头中服务类型域的值，取值范围为 0~255。

#### 【举例】

# 在 ICMP-echo 测试类型下配置探测报文 IP 报文头中服务类型域的值 1。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] tos 1
```

# 在 ICMP 类型的 NQA 模板视图下配置探测报文 IP 报文头中服务类型域的值 1。

```
<Sysname> system-view
[Sysname] nqa template icmp icmptplt
[Sysname-nqatplt-icmp-icmptplt] tos 1
```

### 1.1.72 ttl

**ttl** 命令用来配置探测报文在网络中可以经过的最大跳数。

**undo ttl** 命令用来恢复缺省情况。

#### 【命令】

```
ttl value
undo ttl
```

#### 【缺省情况】

UDP-tracert 类型探测报文在网络中可以经过的最大跳数是 30 跳。其它测试类型下探测报文在网络中可以经过的最大跳数为 20 跳。

### 【视图】

ICMP-echo/TCP/UDP-echo 测试类型视图  
DLSw/DNS/FTP/HTTP/SNMP 测试类型视图  
UDP-tracert 测试类型视图  
ICMP-jitter/UDP-jitter/Voice 测试类型视图  
任意类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

**value**: 对于 UDP-tracert 测试类型，表示允许探测报文填充的最大跳数值；对于其它测试类型，表示探测报文在网络中可以经过的最大跳数。取值范围为 1~255。

### 【使用指导】

配置 **route-option bypass-route** 命令后，探测报文在网络中可以经过的最大跳数为 1，**ttl** 命令不会生效。

配置 UDP-tracert 类型测试时，如果使用 **init-ttl** 命令配置的初始跳数值大于此值，测试将无法启动。

### 【举例】

# 在 ICMP-echo 测试类型下配置探测报文在网络中可以经过的最大跳数为 16 跳。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type icmp-echo
[Sysname-nqa-admin-test-icmp-echo] ttl 16
```

# 在 ICMP 类型的 NQA 模板视图下配置探测报文在网络中可以经过的最大跳数为 16 跳。

```
<Sysname> system-view
[Sysname] nqa template icmp icmptplt
[Sysname-nqatplt-icmp-icmptplt] ttl 16
```

## 1.1.73 type

**type** 命令用来配置当前测试组的测试类型，并进入测试组测试类型视图。

### 【命令】

```
type { dhcp | dlsn | dns | ftp | http | icmp-echo | icmp-jitter | path-jitter |
snmp | tcp | udp-echo | udp-jitter | udp-tracert | voice }
```

### 【缺省情况】

未配置当前测试组的测试类型。

### 【视图】

NQA 测试组视图

### 【缺省用户角色】

network-admin

### 【参数】

**dhcp**: 测试类型为 DHCP。

**dlsw**: 测试类型为 DLSw。

**dns**: 测试类型为 DNS。

**ftp**: 测试类型为 FTP。

**http**: 测试类型为 HTTP。

**icmp-echo**: 测试类型为 ICMP-echo。

**icmp-jitter**: 测试类型为 ICMP-jitter。

**path-jitter**: 测试类型为 Path-jitter。

**snmp**: 测试类型为 SNMP。

**tcp**: 测试类型为 TCP。

**udp-echo**: 测试类型为 UDP-echo。

**udp-jitter**: 测试类型为 UDP-jitter。

**udp-tracert**: 测试类型为 UDP-tracert。

**voice**: 测试类型为 Voice。

### 【使用指导】

一个 NQA 测试组对应一个 NQA 测试类型。当进入 NQA 测试组视图后，需要指定测试类型并配置测试类型相应的属性。当 NQA 测试组指定了一个测试类型后，测试组将与测试类型绑定，若需要重新指定测试类型，请在系统视图下删除测试组之后重新创建测试组并指定测试类型。

### 【举例】

# 配置测试组的测试类型为 FTP 测试，并进入测试组测试类型视图。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type ftp
[Sysname-nqa-admin-test-ftp]
```

## 1.1.74 url

**url** 命令用来配置测试访问的网址。

**undo url** 命令用来恢复缺省情况。

### 【命令】

```
url url
undo url
```

### 【缺省情况】

未配置测试访问的网址。

### 【视图】

FTP/HTTP 测试类型视图

FTP/HTTP/HTTPS 类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

*url*: 测试操作访问的目标资源地址，为 1~255 个字符的字符串，区分大小写。*url* 中的主机名部分，由“.”分隔的字符串组成（如 **aabbcc.com**），每个字符串的长度不超过 63 个字符，区分大小写；字符串中可以包含字母、数字、“-”及“\_”，不能出现连续“.”。

- HTTP 测试类型时，*url* 格式为 **http://host/resource** 或 **http://host:port/resource**。
- HTTPS 测试类型时，*url* 格式为 **https://host/resource** 或 **https://host:port/resource**。
- FTP 测试类型时，*url* 格式为 **ftp://host/filename** 或 **ftp://host:port/filename**  
*filename* 取值范围的详细介绍，请参见“基础配置指导”中的“文件系统管理”。

### 【举例】

# 在 HTTP 测试类型下配置 HTTP 测试访问的网址为 **http://www.company.com/index.html**。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type http
[Sysname-nqa-admin-test-http] url http://www.company.com/index.html
```

# 在 HTTP 类型的 NQA 模板视图下配置测试访问的网址为 **http://www.company.com/index.html**。

```
<Sysname> system-view
[Sysname] nqa template http httptplt
[Sysname-nqatplt-http-httptplt] url http://www.company.com/index.html
```

## 1.1.75 username

**username** 命令用来配置登录用户名。

**undo username** 命令用来恢复缺省情况。

### 【命令】

**username** *username*

**undo username**

### 【缺省情况】

未配置登录用户名。

### 【视图】

FTP/HTTP 测试类型视图

FTP/HTTP/HTTPS/RADIUS 类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

*username*: 测试使用的登录用户名，区分大小写。FTP、HTTP 或 HTTPS 登录用户名，为 1~32 个字符的字符串；RADIUS 用户名，为 1~253 个字符的字符串。



### 【举例】

# 在 FTP 测试类型下配置 FTP 登录用户名为 administrator。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type ftp
[Sysname-nqa-admin-test-ftp] username administrator
```

# 在 FTP 类型的 NQA 模板视图下配置 FTP 登录用户名为 administrator。

```
<Sysname> system-view
[Sysname] nqa template ftp ftptplt
[Sysname-nqatplt-ftp-ftptplt] username administrator
```

### 【相关命令】

- operation
- password

## 1.1.76 version

**version** 命令用来配置 HTTP/HTTPS 测试所使用的版本。

**undo version** 命令用来恢复缺省情况。

### 【命令】

```
version { v1.0 | v1.1 }
undo version
```

### 【缺省情况】

HTTP/HTTPS 测试所使用的版本为 1.0。

### 【视图】

HTTP 测试类型视图

HTTP/HTTPS 类型的 NQA 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

**v1.0**: 测试使用的版本为 1.0。

**v1.1**: 测试使用的版本为 1.1。

### 【举例】

# 在 HTTP 测试类型下配置 HTTP 测试使用的版本为 1.1。

```
<Sysname> system-view
[Sysname] nqa entry admin test
[Sysname-nqa-admin-test] type http
[Sysname-nqa-admin-test-http] version v1.1
```

## 1.2 NQA服务器端命令



说明

只有在测试类型为 UDP-jitter、TCP、UDP-echo 和 Voice 时，才需要配置 NQA 服务器。

### 1.2.1 display nqa server

**display nqa server** 命令用来显示 NQA 服务器的状态信息。

**【命令】**

**display nqa server**

**【视图】**

任意视图

**【缺省用户角色】**

network-admin

network-operator

**【举例】**

# 显示 NQA 服务器的状态信息。

<Sysname> display nqa server

NQA server status: Enabled

TCP connect:

IP address	Port	Tos	VPN instance
2.2.2.2	2000	200	-

UDP echo:

IP address	Port	Tos	VPN instance
3.3.3.3	3000	255	-

表1-10 display nqa server 命令输出信息描述

字段	描述命令
NQA server status	NQA服务器状态，包括的取值如下： <ul style="list-style-type: none"><li>Disabled: 未启用 NQA 服务器功能；</li><li>Enabled: 启用了 NQA 服务器功能；</li></ul>
TCP connect	NQA TCP测试中服务器的状态信息
UDP echo	NQA UDP测试中服务器的状态信息
IP address	NQA服务器TCP/UDP监听服务的IP地址
Port	NQA服务器TCP/UDP监听服务的端口号
Tos	NQA服务器TCP/UDP监听服务的回应报文携带的Tos值
VPN instance	（暂不支持）NQA服务器监听的IP地址所属的VPN实例名称。如果NQA服务器监听的是公网IP地址，则显示为-

### 【相关命令】

- `nqa server enable`
- `nqa server tcp-connect`
- `nqa server udp-echo`

## 1.2.2 nqa server enable

`nqa server enable` 命令用来开启 NQA 服务器功能。

`undo nqa server enable` 命令用来关闭 NQA 服务器功能。

### 【命令】

```
nqa server enable
undo nqa server enable
```

### 【缺省情况】

NQA 服务器功能处于关闭状态。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【举例】

```
# 开启 NQA 服务器功能。
<Sysname> system-view
[Sysname] nqa server enable
```

### 【相关命令】

- `display nqa server`
- `nqa server tcp-connect`
- `nqa server udp-echo`

## 1.2.3 nqa server tcp-connect

`nqa server tcp-connect` 命令用来在 NQA 服务器上创建 TCP 监听服务。

`undo nqa server tcp-connect` 命令用来删除指定的 TCP 监听服务。

### 【命令】

```
nqa server tcp-connect ip-address port-number [ tos tos ]
undo nqa server tcp-connect ip-address port-number
```

### 【缺省情况】

不存在 TCP 监听服务

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*ip-address*: NQA 服务器 TCP 监听服务的 IP 地址。

*port-number*: NQA 服务器 TCP 监听服务的端口号，取值范围为 1~65535。

**tos tos**: NQA 服务器应答报文中的 ToS 域的值。取值范围为 0~255，缺省值为 0。

### 【使用指导】

只有在测试类型为 TCP 时，才需在 NQA 服务器上配置此命令。

所配置的 IP 地址及端口号必须与 NQA 客户端的配置一致，且不能与已有的监听服务冲突。

所配置的 IP 地址必须是作为服务器的设备上的接口的 IP 地址，否则配置无效。

建议不要配置 1~1023 之间的端口（知名端口），否则可能导致 NQA 测试失败或该知名端口对应的服务不可用。

### 【举例】

# 创建 IP 地址为 169.254.10.2，端口号为 9000 的 TCP 监听服务。

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

```
[Sysname] nqa server tcp-connect 169.254.10.2 9000
```

### 【相关命令】

- **display nqa server**
- **nqa server enable**

## 1.2.4 nqa server udp-echo

**nqa server udp-echo** 命令用来在 NQA 服务器上创建 UDP 监听服务。

**undo nqa server udp-echo** 命令用来删除指定的 UDP 监听服务。

### 【命令】

**nqa server udp-echo** *ip-address* *port-number* [**tos tos**]

**undo nqa server udp-echo** *ip-address* *port-number*

### 【缺省情况】

不存在 UDP 监听服务

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*ip-address*: NQA 服务器 UDP 监听服务的 IP 地址。

*port-number*: NQA 服务器 UDP 监听服务的端口号，取值范围为 1~65535。

*tos tos*: NQA 服务器应答报文中的 ToS 域的值。取值范围为 0~255，缺省值为 0。

#### 【使用指导】

只有在测试类型为 UDP-jitter、UDP-echo 或 Voice 时，才需在 NQA 服务器上配置此命令。

配置的 IP 地址及端口号必须与 NQA 客户端的配置一致，且不能与已有的监听服务冲突。

所配置的 IP 地址必须是作为服务器的设备上的接口的 IP 地址，否则配置无效。

建议不要配置 1~1023 之间的端口（知名端口），否则可能导致 NQA 测试失败或该知名端口对应的服务不可用。

#### 【举例】

# 创建 IP 地址为 169.254.10.2、端口号为 9000 的 UDP 监听服务。

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

```
[Sysname] nqa server udp-echo 169.254.10.2 9000
```

#### 【相关命令】

- **display nqa server**
- **nqa server enable**

# 目 录

1 NTP.....	1-1
1.1 NTP配置命令.....	1-1
1.1.1 display ntp-service ipv6 sessions .....	1-1
1.1.2 display ntp-service sessions.....	1-5
1.1.3 display ntp-service status .....	1-9
1.1.4 display ntp-service trace.....	1-11
1.1.5 ntp-service acl.....	1-12
1.1.6 ntp-service authentication enable .....	1-13
1.1.7 ntp-service authentication-keyid.....	1-14
1.1.8 ntp-service broadcast-client .....	1-16
1.1.9 ntp-service broadcast-server .....	1-16
1.1.10 ntp-service dscp .....	1-17
1.1.11 ntp-service enable.....	1-18
1.1.12 ntp-service inbound enable .....	1-18
1.1.13 ntp-service ipv6 acl.....	1-19
1.1.14 ntp-service ipv6 dscp .....	1-20
1.1.15 ntp-service ipv6 inbound enable .....	1-21
1.1.16 ntp-service ipv6 multicast-client.....	1-21
1.1.17 ntp-service ipv6 multicast-server .....	1-22
1.1.18 ntp-service ipv6 source .....	1-23
1.1.19 ntp-service ipv6 unicast-peer .....	1-24
1.1.20 ntp-service ipv6 unicast-server.....	1-25
1.1.21 ntp-service max-dynamic-sessions .....	1-27
1.1.22 ntp-service multicast-client .....	1-28
1.1.23 ntp-service multicast-server.....	1-28
1.1.24 ntp-service refclock-master .....	1-29
1.1.25 ntp-service reliable authentication-keyid.....	1-30
1.1.26 ntp-service source.....	1-31
1.1.27 ntp-service unicast-peer .....	1-32
1.1.28 ntp-service unicast-server .....	1-33
2 SNTP.....	2-1
2.1 SNTP配置命令.....	2-1

2.1.1 display sntp ipv6 sessions .....	2-1
2.1.2 display sntp sessions.....	2-2
2.1.3 sntp authentication enable .....	2-2
2.1.4 sntp authentication-keyid.....	2-3
2.1.5 sntp enable .....	2-5
2.1.6 sntp ipv6 unicast-server .....	2-5
2.1.7 sntp reliable authentication-keyid .....	2-6
2.1.8 sntp unicast-server.....	2-7

# 1 NTP

## 1.1 NTP配置命令

NTP 功能中所指的“接口”为三层接口。

### 1.1.1 display ntp-service ipv6 sessions

**display ntp-service ipv6 sessions** 命令用来显示 NTP 服务的所有 IPv6 会话信息。

#### 【命令】

**display ntp-service ipv6 sessions [ verbose ]**

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

**verbose:** 显示 NTP 服务的所有 IPv6 会话的详细信息。如果不指定该参数，则只显示所有 IPv6 会话的简要信息。

#### 【举例】

# 显示 NTP 服务的所有 IPv6 会话的简要信息。

```
<Sysname> display ntp-service ipv6 sessions
```

```
Notes: 1 source(master), 2 source(peer), 3 selected, 4 candidate, 5 configured.
```

```
Source:    [125]3000::32
```

```
Reference: 127.127.1.0
```

```
Clock stratum: 2
```

```
Reachabilities: 1
```

```
Poll interval: 64
```

```
Last receive time: 6
```

```
Offset: -0.0
```

```
Roundtrip delay: 0.0
```

```
Dispersion: 0.0
```

```
Total sessions: 1
```



表1-1 display ntp-service ipv6 sessions 命令显示信息描述表

字段	描述
[12345]	<p>五个数字所带表的字段含义如下：</p> <ul style="list-style-type: none"> <li>1：系统选中的时间服务器，即当前与设备进行时间同步的时间服务器</li> <li>2：该时间服务器的时钟层数小于等于 15</li> <li>3：该时间服务器的时钟通过了时钟选择算法</li> <li>4：该时间服务器的时钟为候选的时钟</li> <li>5：该时间服务器是通过配置命令指定的</li> </ul>
Source	时间服务器的IPv6地址。若该字段显示为::，表示时间服务器的IPv6地址尚未解析成功
Reference	<p>时间服务器的参考时钟ID</p> <p>当参考时钟为本地时钟时，本字段的显示情况和Clock stratum字段的取值有关：</p> <ul style="list-style-type: none"> <li>当 Clock stratum 字段为 0 或 1 时，本字段显示为 LOCL</li> <li>当 Clock stratum 字段为其他值时，本字段显示为 IPv6 地址前 32 位的 MD5 摘要值，摘要信息按照点分十进制形式显示</li> </ul> <p>当参考时钟为网络中其他设备的时钟时，本字段显示为IPv6地址前32位的MD5摘要值，摘要信息按照点分十进制形式显示。若该字段显示为INIT，表示本地设备还未与时间服务器建立连接</p>
Clock stratum	<p>时间服务器的时钟层数</p> <p>时钟层数决定了时钟的准确度，取值范围为1~16，层数取值越小，时钟的准确度最高，层数为16的时钟处于未同步状态</p>
Reachabilities	时间服务器的可达性计数，0表示时间服务器不可达
Poll interval	轮询间隔，即两个连续NTP报文之间的时间间隔，单位为秒
Last receive time	<p>最近一次接收到NTP报文或更新本地时间到当前时间的时间间隔</p> <p>缺省单位为秒；如果时间间隔大于2048秒，则显示为分钟m；如果时间间隔大于300分钟，则显示为小时h；如果时间间隔大于96小时，则显示为天d；如果时间间隔大于999天，则显示为年y；如果最近一次接收到NTP报文或更新本地时间比当前时间晚，则显示为“-”</p>
Offset	系统时钟相对于参考时钟的时钟偏移，单位为毫秒
Roundtrip delay	本地设备到时间服务器的往返时延，单位为毫秒
Dispersion	系统时钟相对于参考时钟的最大误差，单位为毫秒
Total sessions	总的会话数目

# 显示 NTP 服务的所有 IPv6 会话的详细信息。

```
<Sysname> display ntp-service ipv6 sessions verbose
```

```
Clock source: 1::1
Session ID: 36144
Clock stratum: 16
Clock status: configured, insane, valid, unsynced
Reference clock ID: INIT
Local mode: sym_active, local poll interval: 6
Peer mode: unspec, peer poll interval: 10
```

```

Offset: 0.0000ms, roundtrip delay: 0.0000ms, dispersion: 15937ms
Root roundtrip delay: 0.0000ms, root dispersion: 0.0000ms
Reachabilities:0, sync distance: 15.938
Precision: 2^-19, version: 4, source interface: Not specified
Reftime: 00000000.00000000 Thu, Feb 7 2036 6:28:16.000
Orgtime: d17cbb21.0f318106 Tue, May 17 2011 9:15:13.059
Rcvtime: 00000000.00000000 Thu, Feb 7 2036 6:28:16.000
Xmttime: 00000000.00000000 Thu, Feb 7 2036 6:28:16.000
Roundtrip delay samples: 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
Offset samples: 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
Filter order: 0 1 2 3 4 5 6 7

```

Total sessions: 1

表1-2 display ntp-service ipv6 sessions verbose 命令显示信息描述表

字段	描述
Clock source	时间服务器的IPv6地址。若该字段显示为::, 表示时间服务器的IPv6地址尚未解析成功
Session ID	会话ID
Clock stratum	时间服务器的时钟层数 时钟层数决定了时钟的准确度, 取值范围为1~16, 层数取值越小, 表示时钟的准确度最高, 层数为16的时钟处于未同步状态
Clock status	会话的状态, 该字段的取值及含义为: <ul style="list-style-type: none"> <li>configured: 表示该会话是配置命令所建立的</li> <li>dynamic: 表示该会话是动态生成的</li> <li>master: 表示该会话对应的时间服务器是当前系统的主时间服务器</li> <li>selected: 表示该会话对应时间服务器的时钟通过了时钟选择算法</li> <li>candidate: 表示该会话对应时间服务器的时钟为候选时钟</li> <li>sane: 表示该会话对应的时间服务器通过身份验证, 该时间服务器的时钟将作为参考时钟</li> <li>insane: 表示该会话对应的时间服务器未通过身份验证, 或该时间服务器通过身份验证但其时钟不作为参考时钟</li> <li>valid: 表示该会话对应的时间服务器是有效的 (通过验证、处于同步状态、层数有效、根延时/离差未越界等)</li> <li>invalid: 表示该会话对应的时间服务器是无效的</li> <li>unsynced: 表示该会话对应时间服务器的时钟未同步或层数非法</li> </ul>
Reference clock ID	时间服务器的参考时钟ID 当参考时钟为本地时钟时, 本字段的显示情况和Clock stratum字段的取值有关: <ul style="list-style-type: none"> <li>当 Clock stratum 字段为 0 或 1 时, 本字段显示为 LOCL</li> <li>当 Clock stratum 字段为其他值时, 本字段显示为 IPv6 地址前 32 位的 MD5 摘要值, 摘要信息按照点分十进制形式显示</li> </ul> 当参考时钟为网络中其他设备的时钟时, 本字段显示为IPv6地址前32位的MD5摘要值, 摘要信息按照点分十进制形式显示。若该字段显示为INIT, 表示本地设备还未与时间服务器建立连接

字段	描述
Local mode	本地设备的工作模式，取值包括： <ul style="list-style-type: none"> <li>• <b>unspec</b>: 未指定模式</li> <li>• <b>sym_active</b>: 主动对等体模式</li> <li>• <b>sym_passive</b>: 被动对等体模式</li> <li>• <b>client</b>: 客户端模式</li> <li>• <b>server</b>: 服务器模式</li> <li>• <b>broadcast</b>: 广播服务器模式或组播服务器模式</li> <li>• <b>bclient</b>: 广播客户端模式或组播客户端模式</li> </ul>
local poll interval	本地设备的轮询间隔，显示的是2的次幂数，单位为秒，比如6表示轮询间隔为2的6次幂，即64s
Peer mode	对端设备的工作模式，取值包括： <ul style="list-style-type: none"> <li>• <b>unspec</b>: 未指定模式</li> <li>• <b>sym_active</b>: 主动对等体模式</li> <li>• <b>sym_passive</b>: 被动对等体模式</li> <li>• <b>client</b>: 客户端模式</li> <li>• <b>server</b>: 服务器模式</li> <li>• <b>broadcast</b>: 广播服务器模式或组播服务器模式</li> <li>• <b>bclient</b>: 广播客户端模式或组播客户端模式</li> </ul>
peer poll interval	对端设备的轮询间隔，显示的是2的次幂数，单位为秒，比如6表示轮询间隔为2的6次幂，即64s
Offset	系统时钟相对于参考时钟的时钟偏移，单位为毫秒
roundtrip delay	本地设备到时间服务器的往返时延，单位为毫秒
dispersion	系统时钟相对于参考时钟的最大误差，单位为毫秒
Root roundtrip delay	本地设备到主时间服务器的往返时延，单位为毫秒
root dispersion	系统时钟相对主参考时钟的最大误差，单位为毫秒
Reachabilities	时间服务器的可达性计数，0表示时间服务器不可达
sync distance	表示相对上一级时间服务器的同步距离，由误差disper和往返时延delay计算而来，单位为秒
Precision	系统时钟的精度
version	NTP版本，取值为1~4
source interface	源接口，未指定源接口时，此字段显示为Not specified
Reftime	NTP报文中的参考时间戳
Orgtime	NTP报文中的起始时间戳
Rcvtime	NTP报文的接收时间戳
Xmttime	NTP报文的发送时间戳
Roundtrip delay samples	本地设备到时间服务器往返时延的抽样值

字段	描述
Offset samples	相对于参考时钟的时钟偏移的抽样值
Filter order	样本信息排序
Reference clock status	本地时钟的工作状态，只有通过 <b>ntp-service refclock-master</b> 命令设置本地时钟作为参考时钟时，才会显示该字段 当本地时钟的reach值等于255时，该字段取值为working normally；否则，该字段取值为working abnormally
Total sessions	总的会话数目

### 1.1.2 display ntp-service sessions

**display ntp-service sessions** 命令用来显示 NTP 服务的所有 IPv4 会话信息。

#### 【命令】

**display ntp-service sessions [ verbose ]**

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

**verbose**: 显示 NTP 服务的所有 IPv4 会话的详细信息。如果不指定该参数，则只显示所有会话的简要信息。

#### 【使用指导】

设备作为 NTP 广播服务器或 NTP 组播服务器时，在设备上执行 **display ntp-service sessions** 命令不会显示与该广播服务器或组播服务器对应的 NTP 服务的 IPv4 会话信息，但是这些会话会统计在总的会话数中。

#### 【举例】

# 显示 NTP 服务的所有 IPv4 会话的简要信息。

```
<Sysname> display ntp-service sessions
      source           reference      stra reach poll  now offset  delay disper
*****
[12345]LOCAL(0)        LOCL              0      1   64    - 0.0000 0.0000 7937.9
      [5]0.0.0.0        INIT              16     0   64    - 0.0000 0.0000 0.0000
Notes: 1 source(master), 2 source(peer), 3 selected, 4 candidate, 5 configured.
Total sessions: 1
```

表1-3 display ntp-service sessions 命令显示信息描述表

字段	描述
source	参考时钟为本地时钟时，显示为LOCAL( <i>number</i> )，表示本地时钟的地址为127.127.1. <i>number</i> ，其中 <i>number</i> 为NTP的进程号，取值范围为0~3 参考时钟为网络中其他设备的时钟时，显示为时间服务器的IP地址。若该字段显示为0.0.0.0，表示时间服务器的IP地址尚未解析成功
reference	时间服务器的参考时钟ID 当参考时钟为本地时钟时，本字段的显示情况和stra字段的取值有关： <ul style="list-style-type: none"> <li>当 <b>stra</b> 字段为 0 或 1 时，本字段将显示为 LOCL</li> <li>当 <b>stra</b> 字段为其他值时，本字段将显示为本地时钟的 IP 地址</li> </ul> 当参考时钟为网络中其他设备的时钟时，本字段显示为该设备的IP地址，若该设备为IPv6设备，则本字段显示为该设备的IPv6地址前32位的MD5摘要值，摘要信息按照点分十进制形式显示。若该字段显示为INIT，表示本地设备还未与时间服务器建立连接
stra	时间服务器的时钟层数 时钟层数决定了时钟的准确度，取值范围为1~16，层数取值越小，表示时钟的准确度最高，层数为16的时钟处于未同步状态
reach	时间服务器的可达性计数，0表示时间服务器不可达
poll	轮询间隔，即两个连续NTP报文之间的时间间隔，单位为秒
now	最近一次接收到NTP报文或更新本地时间到当前时间的时间间隔 缺省单位为秒；如果时间间隔大于2048秒，则显示为分钟m；如果时间间隔大于300分钟，则显示为小时h；如果时间间隔大于96小时，则显示为天d；如果时间间隔大于999天，则显示为年y；如果最近一次接收到NTP报文或更新本地时间比当前时间晚，则显示为“-”
offset	系统时钟相对于参考时钟的时钟偏移，单位为毫秒
delay	本地设备到时间服务器的往返时延，单位为毫秒
disper	系统时钟相对于参考时钟的最大误差，单位为毫秒
[12345]	1: 系统选中的时间服务器，即当前与设备进行时间同步的时间服务器 2: 该时间服务器的时钟层数小于等于15 3: 该时间服务器的时钟通过了时钟选择算法 4: 该时间服务器的时钟为候选时钟 5: 该时间服务器的时钟是配置命令指定的
Total sessions	总的会话数目

# 显示 NTP 服务的所有 IPv4 会话的详细信息。

```
<Sysname> display ntp-service sessions verbose
Clock source: 192.168.1.40
Session ID: 35888
Clock stratum: 2
Clock status: configured, master, sane, valid
Reference clock ID: 127.127.1.0
Local mode: client, local poll interval: 6
Peer mode: server, peer poll interval: 6
Offset: 0.2862ms, roundtrip delay: 3.2653ms, dispersion: 4.5166ms
```

```

Root roundtrip delay: 0.0000ms, root dispersion: 10.910ms
Reachabilities:31, sync distance: 0.0194
Precision: 2^-19, version: 3, source interface: Not specified
Reftime: d17cbba5.1473dele Tue, May 17 2011 9:17:25.079
Orgtime: 00000000.00000000 Thu, Feb 7 2036 6:28:16.000
Rcvtime: d17cbbc0.b1959a30 Tue, May 17 2011 9:17:52.693
Xmttime: d17cbbc0.b1959a30 Tue, May 17 2011 9:17:52.693
Roundtrip delay samples: 0.007 0.010 0.006 0.011 0.010 0.005 0.007 0.003
Offset samples: 5629.55 3913.76 5247.27 6526.92 31.99 148.72 38.27 0.29
Filter order: 7 5 2 6 0 4 1 3

Total sessions: 1

```

表1-4 display ntp-service sessions verbose 命令显示信息描述表

字段	描述
Clock source	时间服务器的IP地址。若该字段显示为0.0.0.0，表示时间服务器的IP地址尚未解析成功
Session ID	会话ID
Clock stratum	时间服务器的时钟层数 时钟层数决定了时钟的准确度，取值范围为1~16，层数取值越小，表示时钟的准确度越高，层数为16的时钟处于未同步状态
Clock status	会话的状态，该字段的取值及含义为： <ul style="list-style-type: none"> <li>configured: 表示该会话是配置命令所建立的</li> <li>dynamic: 表示该会话是动态生成的</li> <li>master: 表示该会话对应的时间服务器是当前系统的主时间服务器</li> <li>selected: 表示该会话对应时间服务器的时钟通过了时钟选择算法</li> <li>candidate: 表示该会话对应时间服务器的时钟为候选时钟</li> <li>sane: 表示该会话对应的时间服务器通过身份验证，该时间服务器的时钟将作为参考时钟</li> <li>insane: 表示该会话对应的时间服务器未通过身份验证，或该时间服务器通过身份验证但其时钟不作为参考时钟</li> <li>valid: 表示该会话对应的时间服务器是有效的（通过验证、处于同步状态、层数有效、根延时/误差未越界等）</li> <li>invalid: 表示该会话对应的时间服务器是无效的</li> <li>unsynced: 表示该会话对应时间服务器的时钟未同步或层数非法</li> </ul>
Reference clock ID	时间服务器的参考时钟ID 当参考时钟为本地时钟时，本字段的显示情况和Clock stratum字段的取值有关： <ul style="list-style-type: none"> <li>当 Clock stratum 字段取值为 0 或 1 时，本字段将显示为 LOCL；</li> <li>当 Clock stratum 字段取值为其他值时，本字段将显示为本地时钟的 IP 地址</li> </ul> 当参考时钟为网络中其他设备的时钟时，本字段显示为该设备的IP地址，若该设备为IPv6设备，则本字段显示为该设备的IPv6地址前32位的MD5摘要值，摘要信息按照点分十进制形式显示。若该字段显示为INIT，表示本地设备还未与时间服务器建立连接

字段	描述
Local mode	本地设备的工作模式，取值包括： <ul style="list-style-type: none"> <li>• <b>unspec</b>: 未指定模式</li> <li>• <b>sym_active</b>: 主动对等体模式</li> <li>• <b>sym_passive</b>: 被动对等体模式</li> <li>• <b>client</b>: 客户端模式</li> <li>• <b>server</b>: 服务器模式</li> <li>• <b>broadcast</b>: 广播服务器模式或组播服务器模式</li> <li>• <b>bclient</b>: 广播客户端模式或组播客户端模式</li> </ul>
local poll interval	本地设备的轮询间隔，显示的是2的次幂数，单位为秒，比如6表示轮询间隔为2的6次幂，即64s
Peer mode	对端设备的工作模式，取值包括： <ul style="list-style-type: none"> <li>• <b>unspec</b>: 未指定模式</li> <li>• <b>sym_active</b>: 主动对等体模式</li> <li>• <b>sym_passive</b>: 被动对等体模式</li> <li>• <b>client</b>: 客户端模式</li> <li>• <b>server</b>: 服务器模式</li> <li>• <b>broadcast</b>: 广播服务器模式或组播服务器模式</li> <li>• <b>bclient</b>: 广播客户端模式或组播客户端模式</li> </ul>
peer poll interval	对端设备的轮询间隔，显示的是2的次幂数，单位为秒，比如6表示轮询间隔为2的6次幂，即64s
Offset	系统时钟相对于参考时钟的时钟偏移，单位为毫秒
roundtrip delay	本地设备到时间服务器的往返时延，单位为毫秒
dispersion	系统时钟相对于参考时钟的最大误差
Root roundtrip delay	本地设备到主时间服务器的往返时延，单位为毫秒
root dispersion	系统时钟相对主参考时钟的最大误差，单位为毫秒
Reachabilities	时间服务器的可达性计数，0表示时间服务器不可达
sync distance	表示相对上一级时间服务器的同步距离，由误差disper和往返时延delay计算而来，单位为秒
Precision	系统时钟的精度
version	NTP版本，取值为1~4
source interface	源接口，未指定源接口时，此字段显示为Not specified
Reftime	NTP报文中的参考时间戳
Orgtime	NTP报文中的起始时间戳
Rcvtime	NTP报文的接收时间戳
Xmttime	NTP报文的发送时间戳
Roundtrip delay samples	本地设备到时间服务器往返时延的抽样值

字段	描述
Offset samples	相对于参考时钟的时钟偏移的抽样值
Filter order	抽样信息排序
Reference clock status	本地时钟的工作状态，只有通过 <code>ntp-service refclock-master</code> 命令设置本地时钟作为参考时钟时，才会显示该字段 当本地时钟的reach值等于255时，该字段取值为working normally；否则，该字段取值为working abnormally
Total sessions	总的会话数目

### 1.1.3 display ntp-service status

`display ntp-service status` 命令用来显示 NTP 服务的状态信息。

#### 【命令】

`display ntp-service status`

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【举例】

# 时间已同步时，显示 NTP 服务的状态信息。

```
<Sysname> display ntp-service status
Clock status: synchronized
Clock stratum: 2
System peer: LOCAL(0)
Local mode: client
Reference clock ID: 127.127.1.0
Leap indicator: 00
Clock jitter: 0.000977 s
Stability: 0.000 pps
Clock precision: 2^-19
Root delay: 0.00000 ms
Root dispersion: 3.96367 ms
Reference time: d0c5fc32.92c70b1e Wed, Dec 29 2010 18:28:02.573
System poll interval: 256 s
```

# 时间未同步时，显示 NTP 服务的状态信息。

```
<Sysname> display ntp-service status
Clock status: unsynchronized
Clock stratum: 16
Reference clock ID: none
Clock jitter: 0.000000 s
```



```

Stability: 0.000 pps
Clock precision: 2^-19
Root delay: 0.00000 ms
Root dispersion: 0.00002 ms
Reference time: d0c5fc32.92c70b1e Wed, Dec 29 2010 18:28:02.573
System poll interval: 8 s

```

表1-5 display ntp-service status 命令显示信息描述表

字段	描述
Clock status	<p>系统时间的状态，取值为：</p> <ul style="list-style-type: none"> <li>• <b>synchronized</b>: 系统时间已同步</li> <li>• <b>unsynchronized</b>: 系统时间未同步</li> </ul>
Clock stratum	系统时钟的层数
System peer	系统时钟选中的时间服务器的IP地址
Local mode	<p>相对于选中的时间服务器，本地设备的工作模式，取值包括：</p> <ul style="list-style-type: none"> <li>• <b>unspec</b>: 未指定模式</li> <li>• <b>sym_active</b>: 主动对等体模式</li> <li>• <b>sym_passive</b>: 被动对等体模式</li> <li>• <b>client</b>: 客户端模式</li> <li>• <b>server</b>: 服务器模式</li> <li>• <b>broadcast</b>: 广播服务器模式或组播服务器模式</li> <li>• <b>bclient</b>: 广播客户端模式或组播客户端模式</li> </ul>
Reference clock ID	<p>参考时钟ID</p> <p><b>1. 对于 IPv4 NTP 服务器：</b></p> <p>本地设备从远程时间服务器获取时间同步时，表示远程服务器的IP地址</p> <p>本地设备从本地时钟获取时间同步时，表示本地时钟的标识：</p> <ul style="list-style-type: none"> <li>• 本地时钟的层数为 1 时，显示为 <b>LOCL</b></li> <li>• 本地时钟的层数为其他值时，显示为本地时钟的 IP 地址</li> </ul> <p><b>2. 对于 IPv6 NTP 服务器：</b></p> <p>本地设备从远程时间服务器获取时间同步时，表示远程服务器的IPv6地址前 32位的MD5摘要值</p> <p>本地设备从本地时钟获取时间同步时，表示本地时钟的标识：</p> <ul style="list-style-type: none"> <li>• 本地时钟的层数为 1 时，显示为 <b>LOCL</b></li> <li>• 本地时钟的层数为其他值时，显示为本地时钟的 IPv6 地址前 32 位的 MD5 摘要值</li> </ul>
Leap indicator	<p>告警状态，取值包括：</p> <ul style="list-style-type: none"> <li>• <b>00</b>: 正常状态</li> <li>• <b>01</b>: 闰秒标志，表示一天中的最后一分钟有 61 秒</li> <li>• <b>10</b>: 闰秒标志，表示一天中的最后一分钟有 59 秒</li> <li>• <b>11</b>: 时间未被同步的告警状态</li> </ul>
Clock jitter	系统时钟相对于参考时钟的偏移量，单位为秒

字段	描述
Stability	时钟频率的稳定性，取值越小，时钟频率越稳定
Clock precision	系统时钟的精度
Root delay	本地设备到主时间服务器的往返时延，单位为毫秒
Root dispersion	系统时钟相对主参考时钟的最大误差，单位为毫秒
Reference time	参考时间戳
System poll interval	系统轮询时间间隔，单位为秒

#### 1.1.4 display ntp-service trace

**display ntp-service trace** 命令用来显示从本地设备回溯到主时间服务器的各个 NTP 时间服务器的简要信息。

##### 【命令】

**display ntp-service trace** [ **source** *interface-type interface-number* ]

##### 【视图】

任意视图

##### 【缺省用户角色】

network-admin  
network-operator

##### 【参数】

**source interface-type interface-number**: 指定回溯主时间服务器时发送 NTP 报文的源接口，*interface-type interface-number* 表示接口类型和接口编号。本地设备向时间服务器发送 NTP 报文时，报文的源地址为指定源接口的主 IPv4 地址或接口的 IPv6 地址；如果 NTP 时间服务器地址是链路本地地址时，报文的源地址为报文出接口的链路本地地址。如果不指定本参数，则以报文发送接口作为回溯主时间服务器时发送 NTP 报文的源接口。

##### 【使用指导】

指定源接口回溯主时间服务器时，需要保证主时间服务器已及各个 NTP 时间服务器均和源接口之间路由可达，否则将导致回溯失败。

##### 【举例】

# 显示从本地设备回溯到主时间服务器的各个 NTP 时间服务器的简要信息。

```
<Sysname> display ntp-service trace
Server      127.0.0.1
Stratum     3, jitter 0.000, synch distance 0.0000.
Server      3000::32
Stratum     2, jitter 790.00, synch distance 0.0000.
RefID       127.127.1.0
```

以上信息显示了服务器 127.0.0.1 的同步链：服务器 127.0.0.1 同步到服务器 3000::32，服务器 3000::32 从本地时钟得到同步。

表1-6 display ntp-service trace 命令显示信息描述表

字段	描述
Server	时间服务器的IP地址
Stratum	表示相应服务器的时钟层数
jitter	表示相对上一级时钟的时钟偏差的均方根，单位为秒
synch distance	表示相对上一级时间服务器的同步距离，由误差disper和往返时延delay计算而来，单位为秒
RefID	主时间服务器的标识，主参考时钟的层数为0时，显示为LOCL；为其他值时，显示为主参考时钟的IP地址

#### 【相关命令】

- `ntp-service source`
- `ntp-service unicast-server`
- `ntp-service unicast-peer`
- `ntp-service ipv6 source`
- `ntp-service ipv6 unicast-server`
- `ntp-service ipv6 unicast-peer`

### 1.1.5 ntp-service acl

`ntp-service acl` 命令用来设置对端设备对本地设备 NTP 服务的访问控制权限。

`undo ntp-service acl` 命令用来取消设置的访问控制权限。

#### 【命令】

```
ntp-service { peer | query | server | synchronization } acl ipv4-acl-number
undo ntp-service { peer | query | server | synchronization } [ acl
ipv4-acl-number ]
```

#### 【缺省情况】

对端设备对本地设备 NTP 服务的访问控制权限为 `peer`。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**peer:** 完全访问权限。该权限既允许对端设备向本地设备的时间同步，对本地设备进行控制查询（查询 NTP 的一些状态，比如告警信息、验证状态、时间服务器信息等），同时本地设备也可以向对端设备的时间同步。

**query:** 仅具有控制查询的权限。该权限只允许对端设备对本地设备的 NTP 服务进行控制查询，但是不能向本地设备的时间同步。

**server:** 服务器访问与查询权限。该权限允许对端设备向本地设备的时间同步，对本地设备进行控制查询，但本地设备不会向对端设备的时间同步。

**synchronization:** 仅具有访问服务器的权限。该权限只允许对端设备向本地设备的时间同步，但不能进行控制查询。

**ipv4-acl-number:** 通过编号指定引用的 ACL（Access Control List，访问控制列表）。通过 ACL 过滤的对端设备具有本命令中指定的访问控制权限。*ipv4-acl-number* 为 IPv4 基本或高级 ACL 的编号，取值范围为 2000~2999 和 3000~3999。

### 【使用指导】

NTP 服务的访问控制权限从高到低依次为 **peer**、**server**、**synchronization**、**query**。当设备接收到一个 NTP 服务请求时，会按照权限从高到低的顺序依次进行匹配，第一个匹配的权限为此设备具有的访问控制权限。如果没有匹配的权限，则不允许对端设备与本地设备进行时间同步、对本端进行控制查询，也不允许本端设备与对端设备进行时间同步。

当引用的 ACL 不存在时，任何设备都能访问本地 NTP 服务。当引用的 ACL 下没有配置规则时，任何设备都不能访问本地 NTP 服务。

**ntp-service acl** 命令提供了一种最小限度的安全措施，更安全的方法是进行身份验证。

### 【举例】

# 配置 10.10.0.0/16 网段的对端设备对本地设备具有完全访问权限。

```
<Sysname> system-view
[Sysname] acl basic 2001
[Sysname-acl-ipv4-basic-2001] rule permit source 10.10.0.0 0.0.255.255
[Sysname-acl-ipv4-basic-2001] quit
[Sysname] ntp-service peer acl 2001
```

### 【相关命令】

- **ntp-service authentication enable**
- **ntp-service authentication-keyid**
- **ntp-service reliable authentication-keyid**

## 1.1.6 ntp-service authentication enable

**ntp-service authentication enable** 命令用来开启 NTP 身份验证功能。

**undo ntp-service authentication enable** 命令用来关闭 NTP 身份验证功能。

### 【命令】

```
ntp-service authentication enable
undo ntp-service authentication enable
```

### 【缺省情况】

NTP 身份验证功能处于关闭状态。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【使用指导】

在一些对安全性要求较高的网络中，运行 NTP 协议时需要启用 NTP 身份验证功能。通过客户端和服务端端的身份验证，保证客户端只与通过验证的设备进行时间同步，避免客户端从非法的服务器获得错误的时间同步信息。

开启 NTP 身份验证功能后，还需要设置身份验证密钥，并将其设置为可信密钥，才能正确地进行身份验证。

### 【举例】

# 开启 NTP 身份验证功能。

```
<Sysname> system-view
[Sysname] ntp-service authentication enable
```

### 【相关命令】

- **ntp-service authentication-keyid**
- **ntp-service reliable authentication-keyid**

## 1.1.7 ntp-service authentication-keyid

**ntp-service authentication-keyid** 命令用来设置 NTP 身份验证密钥。

**undo ntp-service authentication-keyid** 命令用来删除指定的 NTP 身份验证密钥。

### 【命令】

```
ntp-service authentication-keyid keyid authentication-mode { hmac-sha-1 |
hmac-sha-256 | hmac-sha-384 | hmac-sha-512 | md5 } { cipher | simple } string
[ acl ipv4-acl-number | ipv6 acl ipv6-acl-number ] *
undo ntp-service authentication-keyid keyid
```

### 【缺省情况】

未设置 NTP 身份验证密钥。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**keyid**: 密钥编号，用来标识身份验证密钥，取值范围为 1~4294967295。

**authentication-mode**: 表示验证算法。

**hmac-sha-1**: 表示采用 HMAC-SHA-1 算法进行身份验证。

**hmac-sha-256**: 表示采用 HMAC-SHA-256 算法进行身份验证。

**hmac-sha-384**: 表示采用 HMAC-SHA-384 算法进行身份验证。

**hmac-sha-512**: 表示采用 HMAC-SHA-512 算法进行身份验证。

**md5:** 表示采用 MD5 算法进行身份验证。

**cipher:** 以密文形式设置密钥。

**simple:** 以明文形式设置密钥，该密钥将以密文形式存储。

**string:** 密钥字符串，区分大小写。明文密钥为 1~32 个字符的字符串，密文密钥为 1~73 个字符的字符串。

**acl ipv4-acl-number:** 对对端设备进行 ACL 过滤。通过 ACL 过滤的对端设备有权在本端使用该密钥 ID 进行身份验证。*ipv4-acl-number* 为 IPv4 基本 ACL 的编号，取值范围为 2000~2999。

**ipv6 acl ipv6-acl-number:** 对对端设备进行 IPv6 ACL 过滤。通过 IPv6 ACL 过滤的对端设备有权在本端使用该密钥 ID 进行身份验证。*ipv6-acl-number* 为 IPv6 基本 ACL 的编号，取值范围为 2000~2999。

### 【使用指导】

在一些对安全性要求较高的网络中，运行 NTP 协议时需要启用身份验证功能。通过客户端和服务器的身份验证，保证客户端只与通过验证的设备进行时间同步，提高了时间同步的安全性。

NTP 协议采用哪个密钥对对端进行身份验证由对端报文中携带的密钥 ID 决定。这会导致如下安全问题：对对端进行身份验证时只关心密钥是否正确而不关心对端是否有权使用该密钥 ID。**acl** 和 **ipv6 acl** 参数用于指定有权在本端使用该密钥 ID 进行身份验证的对端设备。需要注意的是：

- 当本地需要建立或已存在对端的 NTP 会话时，**acl** 和 **ipv6 acl** 参数才会进一步检查对端是否有权在本端使用该密钥 ID。
- 当引用的 ACL 或 IPv6 ACL 不存在时，任何设备都能在本端使用该密钥 ID 进行验证。
- 当引用的 ACL 或 IPv6 ACL 下没有配置规则时，任何设备都不能在本端使用该密钥 ID 进行验证。

客户端和服务端需要配置相同的密钥 ID、验证算法及密钥值，并且保证对端有权在本端使用该密钥 ID 进行身份验证，否则无法实现时间同步。

配置 NTP 验证密钥后，还需要通过 **ntp-service reliable authentication-keyid** 命令将该密钥设置为可信密钥。如果 NTP 验证密钥被指定为可信密钥，删除密钥后，该密钥将自动变为不可信密钥，不必再执行 **undo ntp-service reliable authentication-keyid** 命令。

五种验证算法的安全性从高到低为：HMAC-SHA-512、HMAC-SHA-384、HMAC-SHA-256、HMAC-SHA-1、MD5。

通过重复执行本命令，可以配置多个 NTP 身份验证密钥。设备上最多可以配置 128 个 NTP 身份验证密钥。

### 【举例】

# 设置 MD5 身份验证密钥，密钥 ID 号为 10，密钥为 BetterKey，以明文形式输入。

```
<Sysname> system-view
[Sysname] ntp-service authentication enable
[Sysname] ntp-service authentication-keyid 10 authentication-mode md5 simple BetterKey
```

### 【相关命令】

- **ntp-service authentication enable**
- **ntp-service reliable authentication-keyid**

### 1.1.8 ntp-service broadcast-client

**ntp-service broadcast-client** 命令用来配置设备工作在 NTP 广播客户端模式，并使用当前接口接收 NTP 广播报文。

**undo ntp-service broadcast-client** 命令用来取消 NTP 广播客户端模式的配置。

#### 【命令】

```
ntp-service broadcast-client
undo ntp-service broadcast-client
```

#### 【缺省情况】

未配置 NTP 工作模式。

#### 【视图】

接口视图

#### 【缺省用户角色】

network-admin

#### 【使用指导】

配置设备工作在 NTP 广播客户端模式后，设备将在接口上监听 NTP 广播服务器发送的 NTP 广播报文，根据接收到的报文实现时间同步。

如果在接口上配置了设备工作在广播客户端模式，则建议不要将该接口加入聚合组。如果要将接口加入聚合组，则建议先取消 NTP 广播客户端配置。

#### 【举例】

# 配置设备工作在广播客户端模式，在 VLAN 接口 1 上接收 NTP 广播报文。

```
<Sysname> system-view
[Sysname] interface vlan-interface 1
[Sysname-Vlan-interface1] ntp-service broadcast-client
```

#### 【相关命令】

- **ntp-service broadcast-server**

### 1.1.9 ntp-service broadcast-server

**ntp-service broadcast-server** 命令用来配置设备工作在 NTP 广播服务器模式，并使用当前接口发送 NTP 广播报文。

**undo ntp-service broadcast-server** 命令用来取消 NTP 广播服务器模式的配置。

#### 【命令】

```
ntp-service broadcast-server [ authentication-keyid keyid | version number ]
*
undo ntp-service broadcast-server
```

#### 【缺省情况】

未配置 NTP 工作模式。

## 【视图】

接口视图

## 【缺省用户角色】

network-admin

## 【参数】

**authentication-keyid** *keyid*: 指定向广播客户端发送 NTP 报文时，使用指定的密钥计算报文的摘要。*keyid* 取值范围为 1~4294967295。如果未指定本参数，则本端设备无法同步开启了身份验证功能的广播客户端。

**version** *number*: 指定 NTP 版本号。*number* 取值范围为 1~4，缺省值为 4。

## 【使用指导】

配置设备工作在 NTP 广播服务器模式后，设备将通过该接口周期性地向广播地址 255.255.255.255 发送 NTP 报文。

如果在接口上配置了设备工作在广播服务器模式，则建议不要将该接口加入聚合组。如果要将接口加入聚合组，则建议先取消 NTP 广播服务器配置。

## 【举例】

# 配置设备工作在广播服务器模式，在 VLAN 接口 1 上发送 NTP 广播报文，用 4 号密钥进行加密，设置 NTP 版本号为 4。

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

```
[Sysname] interface vlan-interface 1
```

```
[Sysname-Vlan-interface1] ntp-service broadcast-server authentication-keyid 4 version 4
```

## 【相关命令】

- **ntp-service broadcast-client**

### 1.1.10 ntp-service dscp

**ntp-service dscp** 命令用来配置 NTP 报文的 DSCP 优先级。

**undo ntp-service dscp** 命令用来恢复缺省情况。

## 【命令】

```
ntp-service dscp dscp-value
```

```
undo ntp-service dscp
```

## 【缺省情况】

NTP 报文的 DSCP 优先级为 48。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**dscp-value**: NTP 报文的 DSCP 优先级，取值范围为 0~63。



### 【使用指导】

DSCP 携带在 IP 报文中的 ToS 字段，用来体现报文自身的优先等级，决定报文传输的优先程度。

### 【举例】

# 配置 NTP 报文的 DSCP 优先级为 30。

```
<Sysname> system-view  
[Sysname] ntp-service dscp 30
```

#### 1.1.11 ntp-service enable

**ntp-service enable** 命令用来开启 NTP 服务。

**undo ntp-service enable** 命令用来关闭 NTP 服务。

### 【命令】

```
ntp-service enable  
undo ntp-service enable
```

### 【缺省情况】

NTP 服务处于关闭状态。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【举例】

# 开启 NTP 服务。

```
<Sysname> system-view  
[Sysname] ntp-service enable
```

#### 1.1.12 ntp-service inbound enable

**ntp-service inbound enable** 命令用来开启接口接收 NTP 报文功能。

**undo ntp-service inbound enable** 命令用来关闭接口接收 NTP 报文功能。

### 【命令】

```
ntp-service inbound enable  
undo ntp-service inbound enable
```

### 【缺省情况】

接口接收 NTP 报文功能处于开启状态。

### 【视图】

接口视图

### 【缺省用户角色】

network-admin

## 【使用指导】

如果不允许设备为某个接口对应网段内的对端设备提供时间同步，或不允许设备从某个接口对应网段内的对端设备获得时间同步，则可以在该接口上执行 **undo ntp-service inbound enable** 命令，使该接口关闭接收 NTP 报文功能。

## 【举例】

# 关闭 VLAN 接口 1 接收 NTP 报文功能。

```
<Sysname> system-view
[Sysname] interface vlan-interface 1
[Sysname-Vlan-interface1] undo ntp-service inbound enable
```

### 1.1.13 ntp-service ipv6 acl

**ntp-service ipv6 acl** 命令用来设置对端设备对本地设备 IPv6 NTP 服务的访问控制权限。

**undo ntp-service ipv6 acl** 命令用来取消设置的访问控制权限。

## 【命令】

```
ntp-service ipv6 { peer | query | server | synchronization } acl
ipv6-acl-number
undo ntp-service ipv6 { peer | query | server | synchronization } [ acl
ipv6-acl-number ]
```

## 【缺省情况】

对端设备对本地设备 IPv6 NTP 服务的访问控制权限为 **peer**。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**peer**: 完全访问权限。该权限既允许对端设备向本地设备的时间同步，对本地设备进行控制查询（查询 NTP 的一些状态，比如告警信息、验证状态、时间服务器信息等），同时本地设备也可以向对端设备的时间同步。

**query**: 仅具有控制查询的权限。该权限只允许对端设备对本地设备的 NTP 服务进行控制查询，但是不能向本地设备的时间同步。

**server**: 服务器访问与查询权限。该权限允许对端设备向本地设备的时间同步，对本地设备进行控制查询，但本地设备不会向对端设备的时间同步。

**synchronization**: 仅具有访问服务器的权限。该权限只允许对端设备向本地设备的时间同步，但不能进行控制查询。

**ipv6-acl-number**: 通过编号指定应用的 IPv6 ACL（Access Control List，访问控制列表）。通过 IPv6 ACL 过滤的对端设备具有本命令中指定的访问控制权限。**ipv6-acl-number** 为 IPv6 基本或高级 ACL 的编号，取值范围为 2000～2999 和 3000～3999。

## 【使用指导】

IPv6 NTP 服务的访问控制权限从高到低依次为 **peer**、**server**、**synchronization**、**query**。当设备接收到一个 IPv6 NTP 服务请求时，会按照权限从高到低的顺序依次进行匹配，第一个匹配的权限为此设备具有的访问控制权限。如果没有匹配的权限，则不允许对端设备与本地设备进行时间同步、对本端进行控制查询，也不允许本端设备与对端设备进行时间同步。

当引用的 IPv6 ACL 不存在时，任何设备都能访问本地 IPv6 NTP 服务。当引用的 IPv6 ACL 下没有配置规则时，任何设备都不能访问本地 IPv6 NTP 服务。

**ntp-service ipv6 acl** 命令提供了一种最小限度的安全措施，更安全的方法是进行身份验证。

## 【举例】

# 配置 2001::1 网段的对端设备对本地设备具有完全访问权限。

```
<Sysname> system-view
[Sysname] acl ipv6 basic 2001
[Sysname-acl-ipv6-basic-2001] rule permit source 2001::1 64
[Sysname-acl-ipv6-basic-2001] quit
[Sysname] ntp-service ipv6 peer acl 2001
```

## 【相关命令】

- **ntp-service authentication enable**
- **ntp-service authentication-keyid**
- **ntp-service reliable authentication-keyid**

### 1.1.14 ntp-service ipv6 dscp

**ntp-server ipv6 dscp** 命令用来配置 IPv6 NTP 报文的 DSCP 优先级。

**undo ntp-server ipv6 dscp** 命令用来恢复缺省情况。

## 【命令】

```
ntp-service ipv6 dscp dscp-value
undo ntp-service ipv6 dscp
```

## 【缺省情况】

IPv6 NTP 报文的 DSCP 优先级为 56。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**dscp-value**: IPv6 NTP 报文的 DSCP 优先级，取值范围为 0～63。

## 【使用指导】

DSCP 携带在 IPv6 报文中的 Traffic class 字段，用来体现报文自身的优先等级，决定报文传输的优先程度。

### 【举例】

```
# 配置 IPv6 NTP 报文的 DSCP 优先级为 30。
<Sysname> system-view
[Sysname] ntp-service ipv6 dscp 30
```

## 1.1.15 ntp-service ipv6 inbound enable

**ntp-service ipv6 inbound enable** 命令用来开启接口接收 IPv6 NTP 报文功能。

**undo ntp-service ipv6 inbound enable** 命令用来关闭接口接收 IPv6 NTP 报文功能。

### 【命令】

```
ntp-service ipv6 inbound enable
undo ntp-service ipv6 inbound enable
```

### 【缺省情况】

接口接收 IPv6 NTP 报文功能处于开启状态。

### 【视图】

接口视图

### 【缺省用户角色】

network-admin

### 【使用指导】

如果不允许设备为某个接口对应网段内的对端设备提供时间同步，或不允许设备从某个接口对应网段内的对端设备获得时间同步，则可以在该接口上执行 **undo ntp-service ipv6 inbound enable** 命令，使该接口关闭接收 IPv6 NTP 报文功能。

### 【举例】

```
# 关闭 VLAN 接口 1 接收 IPv6 NTP 报文功能。
<Sysname> system-view
[Sysname] interface vlan-interface 1
[Sysname-Vlan-interface1] undo ntp-service ipv6 inbound enable
```

## 1.1.16 ntp-service ipv6 multicast-client

**ntp-service ipv6 multicast-client** 命令用来配置设备工作在 IPv6 NTP 组播客户端模式，并使用当前接口接收 IPv6 NTP 组播报文。

**undo ntp-service ipv6 multicast-client** 命令用来取消 IPv6 NTP 组播客户端模式的配置。

### 【命令】

```
ntp-service ipv6 multicast-client ipv6-address
undo ntp-service ipv6 multicast-client ipv6-address
```

### 【缺省情况】

未配置 NTP 工作模式。

### 【视图】

接口视图

### 【缺省用户角色】

network-admin

### 【参数】

*ipv6-address*: NTP 报文的 IPv6 组播地址。IPv6 组播客户端和 IPv6 组播服务器上配置的组播地址必须相同。

### 【使用指导】

配置设备工作在 IPv6 NTP 组播客户端模式后，设备将在接口上监听目的地址为指定 IPv6 组播地址的 IPv6 NTP 报文，根据接收到的报文实现时间同步。

如果在接口上配置了设备工作在 IPv6 组播客户端模式，则建议不要将该接口加入聚合组。如果要将接口加入聚合组，则建议先取消 IPv6 NTP 组播客户端配置。

### 【举例】

# 配置设备工作在 IPv6 组播客户端模式，在 VLAN 接口 1 上接收目的地址为组播地址 FF21::1 的 NTP 报文。

```
<Sysname> system-view
[Sysname] interface vlan-interface 1
[Sysname-Vlan-interface1] ntp-service ipv6 multicast-client ff21::1
```

### 【相关命令】

- **ntp-service ipv6 multicast-server**

## 1.1.17 ntp-service ipv6 multicast-server

**ntp-service ipv6 multicast-server** 命令用来配置设备工作在 IPv6 NTP 组播服务器模式，并使用当前接口发送 IPv6 NTP 组播报文。

**undo ntp-service ipv6 multicast-server** 命令用来取消 IPv6 NTP 组播服务器模式的配置。

### 【命令】

```
ntp-service ipv6 multicast-server ipv6-address [ authentication-keyid keyid
| ttl t1-number ] *
undo ntp-service ipv6 multicast-server ipv6-address
```

### 【缺省情况】

未配置 NTP 工作模式。

### 【视图】

接口视图

### 【缺省用户角色】

network-admin

### 【参数】

**ipv6-address**: NTP 报文的 IPv6 组播地址。IPv6 组播客户端和 IPv6 组播服务器上配置的组播地址必须相同。

**authentication-keyid keyid**: 指定向组播客户端发送 NTP 报文时，使用指定的密钥计算报文的摘要。*keyid* 取值范围为 1~4294967295。如果未指定本参数，则本端设备无法同步开启了身份验证功能的组播客户端。

**t1 ttl-number**: 指定组播报文的生存期。*ttl-number* 取值范围为 1~255，缺省值为 16。

### 【使用指导】

配置设备工作在 IPv6 NTP 组播服务器模式后，设备将通过该接口周期性地向指定的 IPv6 组播地址发送 NTP 报文。

如果在接口上配置了设备工作在 IPv6 组播服务器模式，则建议不要将该接口加入聚合组。如果要将接口加入聚合组，则建议先取消 IPv6 NTP 组播服务器配置。

### 【举例】

# 配置设备工作在 IPv6 组播服务器模式，在 VLAN 接口 1 上向 IPv6 组播地址 FF21::1 发送 NTP 报文，用 4 号密钥加密 NTP 报文。

```
<Sysname> system-view
[Sysname] interface vlan-interface 1
[Sysname-Vlan-interface1] ntp-service ipv6 multicast-server ff21::1 authentication-keyid 4
```

### 【相关命令】

- **ntp-service ipv6 multicast-client**

## 1.1.18 ntp-service ipv6 source

**ntp-service ipv6 source** 命令用来指定 IPv6 NTP 报文的源接口。

**undo ntp-service ipv6 source** 命令用来恢复缺省情况。

### 【命令】

```
ntp-service ipv6 source interface-type interface-number
undo ntp-service ipv6 source
```

### 【缺省情况】

未指定 IPv6 NTP 报文的源接口，设备自动选择报文的源 IPv6 地址，具体选择原则请参见 RFC 3484。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

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

## 【使用指导】

如果指定了 IPv6 NTP 报文的源接口，则设备在主动发送 IPv6 NTP 报文时，将采用源接口的 IPv6 地址作为发送报文的源 IPv6 地址，从而保证 IPv6 NTP 应答报文的地址均为该地址。

设备对接收到的 IPv6 NTP 请求报文进行应答时，应答报文的源 IPv6 地址始终为接收到 IPv6 NTP 请求报文的地址。

如果不想让本地设备上其他接口的 IPv6 地址成为应答报文的地址，可以使用本命令。

使用本命令指定 IPv6 NTP 报文的源接口时，需要注意：

- 当 NTP 工作在客户端/服务器模式时，如果在命令 **ntp-service ipv6 unicast-server** 中指定了 IPv6 NTP 报文的源接口，则以 **ntp-service ipv6 unicast-server** 命令指定源接口的为准。
- 当 NTP 工作在对等体模式时，如果在命令 **ntp-service ipv6 unicast-peer** 中指定了 IPv6 NTP 报文的源接口，则以 **ntp-service ipv6 unicast-peer** 命令指定源接口的为准。
- 当 NTP 工作在组播模式时，如果在接口视图下配置了 **ntp-service ipv6 multicast-server** 命令，则 NTP 组播报文的源接口为配置了 **ntp-service ipv6 multicast-server** 命令的接口。
- 如果指定的 NTP 源接口处于 down 状态，则设备不再发送 IPv6 NTP 报文。

## 【举例】

# 配置 IPv6 NTP 报文的源接口为 VLAN 接口 1。

```
<Sysname> system-view
[Sysname] ntp-service ipv6 source vlan-interface 1
```

### 1.1.19 ntp-service ipv6 unicast-peer

**ntp-service ipv6 unicast-peer** 命令用来为设备指定 IPv6 被动对等体。

**undo ntp-service ipv6 unicast-peer** 命令用来删除为设备指定的 IPv6 被动对等体。

## 【命令】

```
ntp-service ipv6 unicast-peer { peer-name | ipv6-address }
[ authentication-keyid keyid | maxpoll maxpoll-interval | minpoll
minpoll-interval | priority | source interface-type interface-number ] *
undo ntp-service ipv6 unicast-peer { peer-name | ipv6-address }
```

## 【缺省情况】

未指定 IPv6 被动对等体。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**peer-name**：被动对等体的主机名，为 1~253 个字符的字符串，不区分大小写。

**ipv6-address**: 被动对等体的 IPv6 地址。该地址只能是一个单播地址，不能为组播地址。

**authentication-keyid keyid**: 指定向对等体发送 NTP 报文时，使用指定的密钥计算报文的摘要。**keyid** 取值范围为 1~4294967295。如果未指定本参数，则本端设备与对等体之间不会进行身份验证。

**maxpoll maxpoll-interval**: 用来配置最大轮询时间间隔。**maxpoll-interval** 取值范围为 4~17，最大轮询时间间隔为 2 的 **maxpoll-interval** 次幂，对应的最大轮询时间间隔的取值范围为  $2^4 \sim 2^{17}$ （即 16~131072）秒。本参数的缺省值为 6，即最大轮询时间间隔为 64 秒。

**minpoll minpoll-interval**: 用来配置最小轮询时间间隔，**minpoll-interval** 取值范围为 4~17，最大轮询时间间隔为 2 的 **maxpoll-interval** 次幂，对应的最大轮询时间间隔的取值范围为  $2^4 \sim 2^{17}$ （即 16~131072）秒。本参数的缺省值为 6，即最小轮询时间间隔为 64 秒。

**priority**: 在同等条件下，优先选择 **ipv6-address** 或 **peer-name** 指定的对等体为同步对等体。

**source interface-type interface-number**: 指定 IPv6 NTP 报文的源接口。如果指定的被动对等体地址不是链路本地地址，则本地设备给对端发送 IPv6 NTP 报文时，报文的源 IPv6 地址为指定源接口的 IPv6 地址。如果指定的被动对等体地址是链路本地地址，则 IPv6 NTP 报文从指定的源接口发送，并且报文的源地址为该接口的链路本地地址。**interface-type interface-number** 为接口类型和接口编号。如果未指定本参数，则设备自动选择报文的源 IPv6 地址，具体选择原则请参见 RFC 3484。

### 【使用指导】

为设备指定 IPv6 被动对等体后，主动对等体和被动对等体的时间可以互相同步。如果双方的时钟都处于同步状态，则层数大的时钟与层数小的时钟的时间同步。

被动对等体的 IPv6 地址为链路本地地址时，必须指定报文的源接口。

主动对等体会按周期与被动对等体进行时间同步，该过程称为轮询。本命令的 **maxpoll** 和 **minpoll** 参数分别用来配置 NTP 对等体模式系统轮询时间间隔的最大值和最小值。选定对等体之后，设备将使用最小轮询间隔进行轮询。当同步时间误差在系统可接受范围内且趋于稳定之后，轮询间隔会逐渐增大直到最大值。当同步时间误差连续超出系统可接受范围多次时，轮询间隔将会减小。

修改轮询间隔不会立即生效，在下次轮询时生效。

### 【举例】

# 配置设备工作在主动对等体模式，被动对等体的 IPv6 地址为 2001::1，NTP 报文的源接口为 VLAN 接口 1。

```
<Sysname> system-view
[Sysname] ntp-service ipv6 unicast-peer 2001::1 source vlan-interface 1
```

### 【相关命令】

- **ntp-service authentication enable**
- **ntp-service authentication-keyid**
- **ntp-service reliable authentication-keyid**

## 1.1.20 ntp-service ipv6 unicast-server

**ntp-service ipv6 unicast-server** 命令用来为设备指定 IPv6 NTP 服务器。

**undo ntp-service ipv6 unicast-server** 命令用来删除为设备指定的 IPv6 NTP 服务器。



## 【命令】

```
ntp-service ipv6 unicast-server { server-name | ipv6-address }  
[ authentication-keyid keyid | maxpoll maxpoll-interval | minpoll  
minpoll-interval | priority | source interface-type interface-number ] *  
undo ntp-service ipv6 unicast-server { server-name | ipv6-address }
```

## 【缺省情况】

未指定 IPv6 NTP 服务器。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**server-name**: NTP 服务器的主机名，为 1~253 个字符的字符串，不区分大小写。

**ipv6-address**: NTP 服务器的 IPv6 地址。该地址只能是一个单播地址，不能为组播地址。

**authentication-keyid keyid**: 指定向 NTP 服务器发送报文时，使用指定的密钥计算报文的摘要。*keyid* 取值范围为 1~4294967295。如果未指定本参数，则本端设备与 NTP 服务器之间不会进行身份验证。

**maxpoll maxpoll-interval**: 用来配置最大轮询时间间隔。*maxpoll-interval* 取值范围为 4~17，最大轮询时间间隔为 2 的 *maxpoll-interval* 次幂，对应的最大轮询时间间隔的取值范围为  $2^4 \sim 2^{17}$ （即 16~131072）秒。本参数的缺省值为 6，即最大轮询时间间隔为 64 秒。

**minpoll minpoll-interval**: 用来配置最小轮询时间间隔，*minpoll-interval* 取值范围为 4~17，最大轮询时间间隔为 2 的 *maxpoll-interval* 次幂，对应的最大轮询时间间隔的取值范围为  $2^4 \sim 2^{17}$ （即 16~131072）秒。本参数的缺省值为 6，即最小轮询时间间隔为 64 秒。

**priority**: 指定在同等条件下，优先选择该服务器。

**source interface-type interface-number**: 指定 IPv6 NTP 报文的源接口。如果指定的 IPv6 NTP 服务器地址不是链路本地地址，则本地设备给服务器发送 IPv6 NTP 报文时，报文的源 IPv6 地址为指定源接口的 IPv6 地址。如果指定的 IPv6 NTP 服务器地址是链路本地地址，则 IPv6 NTP 报文从指定的源接口发送，并且报文的源地址为该接口的链路本地地址。*interface-type interface-number* 为接口类型和接口编号。如果未指定本参数，则设备自动选择报文的源 IPv6 地址，具体选择原则请参见 RFC 3484。

## 【使用指导】

为设备指定 IPv6 NTP 服务器后，设备可以与该服务器的时间同步，但是服务器不会与设备的时间同步。

NTP 服务器的 IPv6 地址为链路本地地址时，必须指定报文的源接口。

NTP 客户端会按周期向 NTP 服务器同步时间，该过程称为轮询。本命令的 **maxpoll** 和 **minpoll** 参数分别用来配置 NTP 客户端服务器模式系统轮询时间间隔的最大值和最小值。选定 NTP 服务器之后，设备将使用最小轮询间隔进行轮询。当同步时间误差在系统可接受范围内且趋于稳定之后，轮询间隔会逐渐增大直到最大值。当同步时间误差连续超出系统可接受范围多次时，轮询间隔将会减小。

修改轮询间隔不会立即生效，在下次轮询时生效。

#### 【举例】

```
# 配置设备的 IPv6 NTP 服务器为 2001::1。
<Sysname> system-view
[Sysname] ntp-service ipv6 unicast-server 2001::1
```

#### 【相关命令】

- **ntp-service authentication enable**
- **ntp-service authentication-keyid**
- **ntp-service reliable authentication-keyid**

### 1.1.21 ntp-service max-dynamic-sessions

**ntp-service max-dynamic-sessions** 命令用来配置 NTP 动态会话的最大数目。

**undo ntp-service max-dynamic-sessions** 命令用来恢复缺省情况。

#### 【命令】

```
ntp-service max-dynamic-sessions number
undo ntp-service max-dynamic-sessions
```

#### 【缺省情况】

NTP 动态会话的最大数目为 100。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*number*: NTP 动态会话的最大数目，取值范围为 0~100。

#### 【使用指导】

同一设备同一时间内存在的会话数目最多为 128 个，其中包括静态会话数和动态会话数。静态会话是用户手动配置 NTP 相关命令而建立的会话；动态会话是 NTP 运行过程中建立的临时会话。

本配置用来限制动态会话的数目，以避免设备上维护过多的动态会话，占用过多的系统资源。

#### 【举例】

```
# 设置 NTP 动态会话的最大数目为 50 个。
<Sysname> system-view
[Sysname] ntp-service max-dynamic-sessions 50
```

#### 【相关命令】

- **display ntp-service sessions**

### 1.1.22 ntp-service multicast-client

**ntp-service multicast-client** 命令用来配置设备工作在 NTP 组播客户端模式，并使用当前接口接收 NTP 组播报文。

**undo ntp-service multicast-client** 命令用来取消 NTP 组播客户端模式的配置。

#### 【命令】

```
ntp-service multicast-client [ ip-address ]  
undo ntp-service multicast-client [ ip-address ]
```

#### 【缺省情况】

未配置 NTP 工作模式。

#### 【视图】

接口视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*ip-address*: NTP 报文的组播 IP 地址，缺省值为 224.0.1.1。组播客户端和组播服务器上配置的组播地址必须相同。

#### 【使用指导】

配置设备工作在 NTP 组播客户端模式后，设备将在接口上监听目的地址为指定组播地址的 NTP 报文，根据接收到的报文实现时间同步。

如果在接口上配置了设备工作在组播客户端模式，则建议不要将该接口加入聚合组。如果要将接口加入聚合组，则建议先取消 NTP 组播客户端配置。

#### 【举例】

# 配置设备工作在组播客户端模式，在 VLAN 接口 1 上接收目的地址为 224.0.1.1 的 NTP 报文。

```
<Sysname> system-view  
[Sysname] interface vlan-interface 1  
[Sysname-Vlan-interface1] ntp-service multicast-client 224.0.1.1
```

#### 【相关命令】

- **ntp-service multicast-server**

### 1.1.23 ntp-service multicast-server

**ntp-service multicast-server** 命令用来配置设备工作在 NTP 组播服务器模式，并使用当前接口发送 NTP 组播报文。

**undo ntp-service multicast-server** 命令用来取消 NTP 组播服务器模式的配置。

#### 【命令】

```
ntp-service multicast-server [ ip-address ] [ authentication-keyid keyid |  
ttl ttl-number | version number ] *  
undo ntp-service multicast-server [ ip-address ]
```

### 【缺省情况】

未配置 NTP 工作模式。

### 【视图】

接口视图

### 【缺省用户角色】

network-admin

### 【参数】

**ip-address**: NTP 报文的组播 IP 地址，缺省值为 224.0.1.1。组播客户端和组播服务器上配置的组播地址必须相同。

**authentication-keyid keyid**: 指定向组播客户端发送 NTP 报文时，使用指定的密钥计算报文的摘要。*keyid* 取值范围为 1~4294967295。如果未指定本参数，则本端设备无法同步开启了身份验证功能的组播客户端。

**t11 ttl-number**: 指定组播报文的生存期。*ttl-number* 取值范围为 1~255，缺省值为 16。

**version number**: 指定 NTP 版本号。*number* 取值范围为 1~4，缺省值为 4。

### 【使用指导】

配置设备工作在 NTP 组播服务器模式后，设备将通过该接口周期性地向指定的组播地址发送 NTP 报文。

如果在接口上配置了设备工作在组播服务器模式，则建议不要将该接口加入聚合组。如果要将接口加入聚合组，则建议先取消 NTP 组播服务器配置。

### 【举例】

# 配置设备工作在组播服务器模式，在 VLAN 接口 1 上发送 NTP 报文，NTP 报文的地址为组播地址 224.0.1.1，用 4 号密钥加密 NTP 报文，并设置 NTP 版本号为 4。

```
<Sysname> system-view
[Sysname] interface vlan-interface 1
[Sysname-Vlan-interface1] ntp-service multicast-server 224.0.1.1 version 4
authentication-keyid 4
```

### 【相关命令】

- **ntp-service multicast-client**

## 1.1.24 ntp-service refclock-master

**ntp-service refclock-master** 命令用来设置本地时钟作为参考时钟。

**undo ntp-service refclock-master** 命令用来取消本地时钟作为参考时钟。

### 【命令】

```
ntp-service refclock-master [ ip-address ] [ stratum ]
undo ntp-service refclock-master [ ip-address ]
```

### 【缺省情况】

设备未采用本地时钟作为参考时钟。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

*ip-address*: 本地时钟的 IP 地址 127.127.1.u。u 的取值范围为 0~3，表示 NTP 的进程号。如果不指定 *ip-address*，则系统默认值是 127.127.1.0。

*stratum*: 本地时钟所处的层数，取值范围为 1~15，缺省值为 8。时钟的层数定义了时钟的准确度，层数取值越小，时钟的准确度越高。

## 【使用指导】

实际网络中，通常将从权威时钟（如原子时钟）获得时间同步的 NTP 服务器的层数设置为 1，并将其作为主时间服务器同步网络中其他设备的时钟。网络中的设备与主时间服务器的 NTP 距离，即 NTP 同步链上 NTP 服务器的数目，决定了设备上时钟的层数。

在某些网络中，例如无法与外界通信的孤立网络，网络中的设备无法与权威时钟进行时间同步。此时，可以从该网络中选择一台时钟较为准确的设备，指定该设备与本地时钟进行时间同步，即采用本地时钟作为参考时钟，使得该设备的时钟处于同步状态。该设备作为时间服务器为网络中的其他设备提供时间同步，从而实现整个网络的时间同步。

请谨慎使用本配置，以免导致网络中设备的时间错误。在执行本命令之前，建议先调整本地系统时间。

## 【举例】

# 设置本地设备时钟作为参考时钟，层数为 2。

```
<Sysname> system-view
[Sysname] ntp-service refclock-master 2
```

### 1.1.25 ntp-service reliable authentication-keyid

**ntp-service reliable authentication-keyid** 命令用来指定已创建的密钥是可信的。

**undo ntp-service reliable authentication-keyid** 命令用来取消可信密钥。

## 【命令】

```
ntp-service reliable authentication-keyid keyid
undo ntp-service reliable authentication-keyid keyid
```

## 【缺省情况】

未指定可信密钥。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

### 【参数】

*keyid*: 密钥编号，取值范围为 1~4294967295。

### 【使用指导】

开启身份验证功能后，客户端只会与提供可信密钥的服务器进行时间同步；如果服务器提供的密钥不是可信的，那么客户端不会与其同步。

配置本命令前，请确认证开关已经打开并且配置了密钥，即保证该密钥的存在性后才能设定它是否可信。如果 NTP 验证密钥被指定为可信密钥，删除密钥后，该密钥将自动变为不可信密钥，不必再执行 **undo ntp-service reliable authentication-keyid** 命令。

本命令可以多次配置，最多可以配置 128 个可信密钥。

### 【举例】

# 开启 NTP 身份验证功能，配置编号为 37 的密钥采用 MD5 算法进行身份验证，密钥值为 BetterKey。

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

```
[Sysname] ntp-service authentication enable
```

```
[Sysname] ntp-service authentication-keyid 37 authentication-mode md5 simple BetterKey
```

# 指定该密钥为可信密钥。

```
[Sysname] ntp-service reliable authentication-keyid 37
```

### 【相关命令】

- **ntp-service authentication enable**
- **ntp-service authentication-keyid**

## 1.1.26 ntp-service source

**ntp-service source** 命令用来指定 NTP 报文的源接口。

**undo ntp-service source** 命令用来恢复缺省情况。

### 【命令】

```
ntp-service source interface-type interface-number
```

```
undo ntp-service source
```

### 【缺省情况】

未指定 NTP 报文的源接口，设备根据路由表查找报文的出接口，并采用出接口的主 IP 地址作为 NTP 报文的源 IP 地址。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*interface-type interface-number*: 接口类型及接口编号。

## 【使用指导】

如果指定了 NTP 报文的源接口，则设备在主动发送 NTP 报文时，将报文的源 IP 地址设置为指定接口的主 IP 地址，从而保证 NTP 应答报文的目地址均为此地址。

设备对接收到的 NTP 请求报文进行应答时，应答报文的源地址始终为接收到 NTP 请求报文的目地址。

如果不想让本地设备上其他接口的 IP 地址成为应答报文的目地址，可以使用本命令。

使用本命令指定 NTP 报文的源接口时，需要注意：

- 当 NTP 工作在客户端/服务器模式时，如果在命令 **ntp-service unicast-server** 中指定了 NTP 报文的源接口，则以 **ntp-service unicast-server** 命令指定的源接口为准。
- 当 NTP 工作在对等体模式时，如果在命令 **ntp-service unicast-peer** 中指定了 NTP 报文的源接口，则以 **ntp-service unicast-peer** 命令指定源接口的为准。
- 当 NTP 工作在组播模式时，如果在接口视图下配置了 **ntp-service multicast-server** 命令，则 NTP 组播报文的源接口为配置了 **ntp-service multicast-server** 命令的接口。
- 当 NTP 工作在广播模式时，如果在接口视图下配置了 **ntp-service broadcast-server** 命令，则 NTP 广播报文的源接口为配置了 **ntp-service broadcast-server** 命令的接口。
- 如果指定的 NTP 源接口处于 down 状态，则设备不再发送 NTP 报文。

## 【举例】

# 配置 NTP 报文的源接口为 VLAN 接口 1。

```
<Sysname> system-view
[Sysname] ntp-service source vlan-interface 1
```

### 1.1.27 ntp-service unicast-peer

**ntp-service unicast-peer** 命令用来为设备指定被动对等体。

**undo ntp-service unicast-peer** 命令用来删除为设备指定的被动对等体。

## 【命令】

```
ntp-service unicast-peer { peer-name | ip-address } [ authentication-keyid keyid | maxpoll maxpoll-interval | minpoll minpoll-interval | priority | source interface-type interface-number | version number ] *
undo ntp-service unicast-peer { peer-name | ip-address }
```

## 【缺省情况】

未指定被动对等体。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**peer-name**: 被动对等体的主机名, 为 1~253 个字符的字符串, 不区分大小写。

**ip-address**: 被动对等体的 IP 地址。该地址只能是一个单播地址, 不能为广播地址、组播地址或本地时钟的 IP 地址。

**authentication-keyid keyid**: 指定向对等体发送 NTP 报文时, 使用指定的密钥计算报文的摘要。*keyid* 取值范围为 1~4294967295。如果未指定本参数, 则本端设备与对等体之间不会进行身份验证。

**maxpoll maxpoll-interval**: 用来配置最大轮询时间间隔。*maxpoll-interval* 取值范围为 4~17, 最大轮询时间间隔为 2 的 *maxpoll-interval* 次幂, 对应的最大轮询时间间隔的取值范围为  $2^4 \sim 2^{17}$  (即 16~131072) 秒。本参数的缺省值为 6, 即最大轮询时间间隔为 64 秒。

**minpoll minpoll-interval**: 用来配置最小轮询时间间隔, *minpoll-interval* 取值范围为 4~17, 最大轮询时间间隔为 2 的 *maxpoll-interval* 次幂, 对应的最大轮询时间间隔的取值范围为  $2^4 \sim 2^{17}$  (即 16~131072) 秒。本参数的缺省值为 6, 即最小轮询时间间隔为 64 秒。

**priority**: 在同等条件下, 优先选择 *ip-address* 或 *peer-name* 指定的对等体为同步对等体。

**source interface-type interface-number**: 指定 NTP 报文的源接口。本地设备给对端发送 NTP 报文时, 报文的源地址为指定源接口的主 IP 地址。*interface-type interface-number* 为接口类型和接口编号。如果未指定本参数, 则根据路由表查找报文的出接口, 并采用出接口的主 IP 地址作为 NTP 报文的源 IP 地址。

**version number**: 指定 NTP 版本号。*number* 取值范围为 1~4, 缺省值为 4。

## 【使用指导】

为设备指定被动对等体后, 主动对等体和被动对等体的时间可以互相同步。如果双方的时钟都处于同步状态, 则层数大的时钟与层数小的时钟的时间同步。

主动对等体会按周期与被动对等体进行时间同步, 该过程称为轮询。本命令的 **maxpoll** 和 **minpoll** 参数分别用来配置 NTP 对等体模式系统轮询时间间隔的最大值和最小值。选定对等体之后, 设备将使用最小轮询间隔进行轮询。当同步时间误差在系统可接受范围内且趋于稳定时, 轮询间隔会逐渐增大直到最大值。当同步时间误差连续超出系统可接受范围多次时, 轮询间隔将会减小。

修改轮询间隔不会立即生效, 在下次轮询时生效。

## 【举例】

# 配置设备工作在主动对等体模式, 被动对等体的 IP 地址为 10.1.1.1, NTP 版本号为 4, NTP 报文的源接口为 VLAN 接口 1。

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

```
[Sysname] ntp-service unicast-peer 10.1.1.1 version 4 source vlan-interface 1
```

## 【相关命令】

- **ntp-service authentication enable**
- **ntp-service authentication-keyid**
- **ntp-service reliable authentication-keyid**

### 1.1.28 ntp-service unicast-server

**ntp-service unicast-server** 命令用来为设备指定 NTP 服务器。



**undo ntp-service unicast-server** 命令用来删除为设备指定的 NTP 服务器。

#### 【命令】

```
ntp-service unicast-server { server-name | ip-address }  
[ authentication-keyid keyid | maxpoll maxpoll-interval | minpoll  
minpoll-interval | priority | source interface-type interface-number |  
version number ] *  
undo ntp-service unicast-server { server-name | ip-address }
```

#### 【缺省情况】

未指定 NTP 服务器。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**server-name**: NTP 服务器的主机名，为 1~253 个字符的字符串，不区分大小写。

**ip-address**: NTP 服务器的 IP 地址。该地址只能是一个单播地址，不能为广播地址、组播地址或本地时钟的 IP 地址。

**authentication-keyid keyid**: 指定向 NTP 服务器发送报文时，使用指定的密钥计算报文的摘要。**keyid** 取值范围为 1~4294967295。如果未指定本参数，则本端设备与 NTP 服务器之间不会进行身份验证。

**maxpoll maxpoll-interval**: 用来配置最大轮询时间间隔。**maxpoll-interval** 取值范围为 4~17，最大轮询时间间隔为 2 的 **maxpoll-interval** 次幂，对应的最大轮询时间间隔的取值范围为  $2^4 \sim 2^{17}$ （即 16~131072）秒。本参数的缺省值为 6，即最大轮询时间间隔为 64 秒。

**minpoll minpoll-interval**: 用来配置最小轮询时间间隔，**minpoll-interval** 取值范围为 4~17，最大轮询时间间隔为 2 的 **maxpoll-interval** 次幂，对应的最大轮询时间间隔的取值范围为  $2^4 \sim 2^{17}$ （即 16~131072）秒。本参数的缺省值为 6，即最小轮询时间间隔为 64 秒。

**priority**: 指定在同等条件下，优先选择该服务器。

**source interface-type interface-number**: 指定 NTP 报文的源接口。本地设备给服务器发送 NTP 报文时，报文的源地址为指定源接口的主 IP 地址。**interface-type interface-number** 为接口类型和接口编号。如果未指定本参数，则根据路由表查找报文的出接口，并采用出接口的主 IP 地址作为 NTP 报文的源 IP 地址。

**version number**: 指定 NTP 版本号。**number** 取值范围为 1~4，缺省值为 4。

#### 【使用指导】

为设备指定 NTP 服务器后，设备可以与该服务器的时间同步，但是服务器不会与设备的时间同步。

NTP 客户端会按周期向 NTP 服务器同步时间，该过程称为轮询。本命令的 **maxpoll** 和 **minpoll** 参数分别用来配置 NTP 客户端服务器模式系统轮询时间间隔的最大值和最小值。选定 NTP 服务器之后，设备将使用最小轮询间隔进行轮询。当同步时间误差在系统可接受范围内且趋于稳定之后，轮询间隔会逐渐增大直到最大值。当同步时间误差连续超出系统可接受范围多次时，轮询间隔将会减小。

修改轮询间隔不会立即生效，在下次轮询时生效。

#### 【举例】

# 配置设备的 NTP 服务器为 10.1.1.1，版本号为 4。

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

```
[Sysname] ntp-service unicast-server 10.1.1.1 version 4
```

#### 【相关命令】

- **ntp-service authentication enable**
- **ntp-service authentication-keyid**
- **ntp-service reliable authentication-keyid**

# 2 SNTP

## 2.1 SNTP配置命令

### 2.1.1 display sntp ipv6 sessions

`display sntp ipv6 sessions` 命令用来显示 SNTP 服务的所有 IPv6 会话信息。

【命令】

`display sntp ipv6 sessions`

【视图】

任意视图

【缺省用户角色】

network-admin  
network-operator

【举例】

```
# 显示 IPv6 SNTP 服务的所有 IPv6 会话信息。
<Sysname> display sntp ipv6 sessions
SNTP server: 2001::1
Stratum: 16
Version: 4
Last receive time: No packet was received.

SNTP server: 2001::100
Stratum: 3
Version: 4
Last receive time: Fri, Oct 21 2011 11:28:28.058 (Synced)
```

表2-1 display sntp ipv6 sessions 命令显示信息描述表

字段	描述
SNTP server	SNTP服务器，即NTP服务器。若该字段显示为::，表示NTP服务器的IPv6地址尚未解析成功
Stratum	时钟的层数 时钟层数决定了时钟的准确度，取值范围为1~16，层数取值越小，表示时钟的准确度越高，层数为16的时钟处于未同步状态
Version	版本号
Last receive time	最后一次接收到SNTP会话消息的时间 <ul style="list-style-type: none"><li>Synced 表示设备的本地时钟从该服务器获得同步</li><li>No packet was received.表示设备未从该服务器接收到 SNTP 会话消息</li></ul>

## 2.1.2 display sntp sessions

**display sntp sessions** 命令用来显示 SNTP 服务的所有 IPv4 会话信息。

### 【命令】

**display sntp sessions**

### 【视图】

任意视图

### 【缺省用户角色】

network-admin  
network-operator

### 【举例】

# 显示 SNTP 服务的所有 IPv4 会话信息。

```
<Sysname> display sntp sessions
SNTP server      Stratum   Version   Last receive time
1.0.1.11         2          4         Tue, May 17 2011  9:11:20.833 (Synced)
```

表2-2 display sntp sessions 命令显示信息描述表

字段	描述
SNTP server	SNTP服务器，即NTP服务器。若该字段显示为0.0.0.0，表示NTP服务器的IP地址尚未解析成功
Stratum	时钟的层数 时钟层数决定了时钟的准确度，取值范围为1~16，层数取值越小，表示时钟的准确度越高，层数为16的时钟处于未同步状态
Version	SNTP版本号
Last receive time	上一次接收到消息的时间，Synced标识本地时钟从该服务器获得同步

## 2.1.3 sntp authentication enable

**sntp authentication enable** 命令用来开启 SNTP 身份验证功能。

**undo sntp authentication enable** 命令用来关闭 SNTP 身份验证功能。

### 【命令】

**sntp authentication enable**  
**undo sntp authentication enable**

### 【缺省情况】

SNTP 身份验证功能处于关闭状态。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【使用指导】

在一些对安全性要求较高的网络中，运行 SNTP 协议时需要启用身份验证功能。通过客户端和服务器的身份验证，保证客户端只与通过验证的服务器进行时间同步，避免客户端从非法的服务器获得错误的时间同步信息。

开启 SNTP 身份验证功能后，还需要设置身份验证密钥，并将其设置为可信密钥，才能正确地进行身份验证。

### 【举例】

# 开启 SNTP 身份验证功能。

```
<Sysname> system-view
[Sysname] sntp authentication enable
```

### 【相关命令】

- **sntp authentication-keyid**
- **sntp reliable authentication-keyid**

## 2.1.4 sntp authentication-keyid

**sntp authentication-keyid** 命令用来设置 SNTP 身份验证密钥。

**undo sntp authentication-keyid** 命令用来删除指定的 SNTP 身份验证密钥。

### 【命令】

```
sntp authentication-keyid keyid authentication-mode { hmac-sha-1 |
hmac-sha-256 | hmac-sha-384 | hmac-sha-512 | md5 } { cipher | simple } string
[ acl ipv4-acl-number | ipv6 acl ipv6-acl-number ] *
undo sntp authentication-keyid keyid
```

### 【缺省情况】

未设置 SNTP 身份验证密钥。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**keyid**: 密钥编号，用来标识身份验证密钥，取值范围为 1~4294967295。

**authentication-mode**: 表示验证算法。

**hmac-sha-1**: 表示采用 HMAC-SHA-1 算法进行身份验证。

**hmac-sha-256**: 表示采用 HMAC-SHA-256 算法进行身份验证。

**hmac-sha-384**: 表示采用 HMAC-SHA-384 算法进行身份验证。

**hmac-sha-512**: 表示采用 HMAC-SHA-512 算法进行身份验证。

**md5:** 表示采用 MD5 算法进行身份验证。

**cipher:** 以密文形式设置密钥。

**simple:** 以明文形式设置密钥，该密钥将以密文形式存储。

**string:** 密钥字符串，区分大小写。明文密钥为 1~32 个字符的字符串，密文密钥为 1~73 个字符的字符串。

**acl ipv4-acl-number:** 对对端设备进行 ACL 过滤。通过 ACL 过滤的对端设备有权在本端使用该密钥 ID 进行身份验证。*ipv4-acl-number* 为 IPv4 基本 ACL 的编号，取值范围为 2000~2999。

**ipv6 acl ipv6-acl-number:** 对对端设备进行 IPv6 ACL 过滤。通过 IPv6 ACL 过滤的对端设备有权在本端使用该密钥 ID 进行身份验证。*ipv6-acl-number* 为 IPv6 基本 ACL 的编号，取值范围为 2000~2999。

### 【使用指导】

在一些对安全性要求较高的网络中，运行 SNTP 协议时需要启用身份验证功能。通过客户端和服务器的身份验证，保证客户端只与通过验证的服务器进行同步，提高了网络安全性。

SNTP 协议采用哪个密钥对对端进行身份验证由对端报文中携带的密钥 ID 决定。这会导致如下安全问题：对对端进行身份验证时只关心密钥是否正确而不关心对端是否有权使用该密钥 ID。**acl** 和 **ipv6 acl** 参数用于指定有权在本端使用该密钥 ID 进行身份验证的对端设备。需要注意的是：

当本地需要建立或已存在对端的 SNTP 会话时，**acl** 和 **ipv6 acl** 参数才会进一步检查对端是否有权在本端使用该密钥 ID。

当引用的 ACL 或 IPv6 ACL 不存在时，任何设备都能在本端使用该密钥 ID 进行验证。

当引用的 ACL 或 IPv6 ACL 下没有配置规则时，任何设备都不能在本端使用该密钥 ID 进行验证。

客户端和服务端上需要配置相同的密钥 ID、验证算法及密钥值，并且保证对端有权在本端使用该密钥 ID 进行身份验证，否则无法实现时间同步。

配置 SNTP 验证密钥后，还需要通过 **sntp reliable authentication-keyid** 命令将该密钥设置为可信密钥。如果 SNTP 验证密钥被指定为可信密钥，删除密钥后，该密钥将自动变为不可信密钥，不必再执行 **undo sntp reliable authentication-keyid** 命令。

五种验证算法的安全性从高到低为：HMAC-SHA-512、HMAC-SHA-384、HMAC-SHA-256、HMAC-SHA-1、MD5。

通过重复执行本命令，可以配置多个 SNTP 身份验证密钥。设备上最多可以配置 128 个 SNTP 身份验证密钥。

### 【举例】

# 设置 MD5 身份验证密钥，密钥 ID 号为 10，密钥为 BetterKey，以明文形式输入。

```
<Sysname> system-view
[Sysname] sntp authentication enable
[Sysname] sntp authentication-keyid 10 authentication-mode md5 simple BetterKey
```

### 【相关命令】

- **sntp authentication enable**
- **sntp reliable authentication-keyid**

### 2.1.5 sntp enable

**sntp enable** 命令用来开启 SNTP 服务。

**undo sntp enable** 命令用来关闭 SNTP 服务。

#### 【命令】

```
sntp enable
undo sntp enable
```

#### 【缺省情况】

SNTP 服务处于关闭状态。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【举例】

```
# 开启 SNTP 服务。
<Sysname> system-view
[Sysname] sntp enable
```

### 2.1.6 sntp ipv6 unicast-server

**sntp ipv6 unicast-server** 命令用来为设备指定 IPv6 NTP 服务器。

**undo sntp ipv6 unicast-server** 命令用来删除为设备指定的 IPv6 NTP 服务器。

#### 【命令】

```
sntp          ipv6          unicast-server          {          server-name          |
ipv6-address  }[ authentication-keyid keyid | source interface-type
interface-number ] *
undo sntp ipv6 unicast-server { server-name | ipv6-address }
```

#### 【缺省情况】

未指定 IPv6 NTP 服务器。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**server-name**: NTP 服务器的主机名，为 1~253 个字符的字符串，不区分大小写。

**ipv6-address**: NTP 服务器的 IPv6 地址。

**authentication-keyid** *keyid*: 指定向 NTP 服务器发送报文时, 使用指定的密钥计算报文的摘要。*keyid* 取值范围为 1~4294967295。如果未指定本参数, 则本端设备与 NTP 服务器之间不会进行身份验证。

**source interface-type interface-number**: 指定 IPv6 NTP 报文的源接口。如果指定的 IPv6 NTP 服务器地址不是链路本地地址, 则本地设备给服务器发送 IPv6 NTP 报文时, 报文的源 IPv6 地址为指定源接口的 IPv6 地址。如果指定的 IPv6 NTP 服务器地址是链路本地地址, 则 IPv6 NTP 报文从指定的源接口发送, 并且报文的源地址为该接口的链路本地地址。*interface-type interface-number* 为接口类型和接口编号。如果未指定本参数, 则设备自动选择报文的源 IPv6 地址, 具体选择原则请参见 RFC 3484。

#### 【使用指导】

为设备指定 IPv6 NTP 服务器后, 设备可以与该服务器进行时间同步。设备的时间获得同步后, 不能作为服务器为其他设备提供时间同步。

#### 【举例】

```
# 配置设备的 NTP 服务器为 2001::1。
<Sysname> system-view
[Sysname] sntp ipv6 unicast-server 2001::1
```

#### 【相关命令】

- **sntp authentication enable**
- **sntp authentication-keyid**
- **sntp reliable authentication-keyid**

### 2.1.7 sntp reliable authentication-keyid

**sntp reliable authentication-keyid** 命令用来配置可信密钥。

**undo sntp reliable authentication-keyid** 命令用来取消可信密钥。

#### 【命令】

```
sntp reliable authentication-keyid keyid
undo sntp reliable authentication-keyid keyid
```

#### 【缺省情况】

未指定可信密钥。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*keyid*: 密钥编号, 取值范围为 1~4294967295。



### 【使用指导】

开启身份验证功能后，客户端只会同步到提供可信密钥的服务器；如果服务器提供的密钥不是可信的，那么客户端不会与其同步。

本命令的使用前提是认证开关已经打开并且配置了密钥，即保证该密钥的存在性后才能设定它是否可信。如果 SNTP 验证密钥被指定为可信密钥，删除密钥后，该密钥将自动变为不可信密钥，不必再执行 **undo sntp reliable authentication-keyid** 命令。

### 【举例】

# 开启 SNTP 身份验证功能，配置编号为 37 的密钥采用 MD5 算法进行身份验证，密钥值为 BetterKey。

```
<Sysname> system-view
[Sysname] sntp authentication enable
[Sysname] sntp authentication-keyid 37 authentication-mode md5 simple BetterKey
```

# 指定该密钥为可信密钥。

```
[Sysname] sntp reliable authentication-keyid 37
```

### 【相关命令】

- **sntp authentication-keyid**
- **sntp authentication enable**

## 2.1.8 sntp unicast-server

**sntp unicast-server** 命令用来为设备指定 NTP 服务器。

**undo sntp unicast-server** 命令用来删除为设备指定的 NTP 服务器。

### 【命令】

```
sntp unicast-server { server-name | ip-address } [ authentication-keyid keyid | source interface-type interface-number | version number ] *
undo sntp unicast-server { server-name | ip-address }
```

### 【缺省情况】

未指定 NTP 服务器。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**server-name**: NTP 服务器的主机名，为 1~253 个字符的字符串，不区分大小写。

**ip-address**: NTP 服务器的 IP 地址。该地址只能是一个单播地址，不能为广播地址、组播地址或本地时钟的 IP 地址。

**authentication-keyid keyid**: 指定向 NTP 服务器发送报文时，使用指定的密钥计算报文的摘要。*keyid* 取值范围为 1~4294967295。如果未指定本参数，则本端设备与 NTP 服务器之间不会进行身份验证。

**source interface-type interface-number:** 指定 NTP 报文的源接口。本地设备给服务器发送 NTP 报文时，报文的源地址为指定源接口的主 IP 地址。*interface-type interface-number* 为接口类型和接口编号。如果未指定本参数，则根据路由表查找报文的出接口，并采用出接口的主 IP 地址作为 NTP 报文的源 IP 地址。

**version number:** 指定 NTP 版本号。*number* 取值范围为 1~4，缺省值为 4。

#### 【使用指导】

为设备指定 NTP 服务器后，设备可以与该服务器进行时间同步。设备的时间获得同步后，不能作为服务器为其他设备提供时间同步。

#### 【举例】

```
# 配置设备的 NTP 服务器为 10.1.1.1，版本号为 4。
<Sysname> system-view
[Sysname] sntp unicast-server 10.1.1.1 version 4
```

#### 【相关命令】

- **sntp authentication enable**
- **sntp authentication-keyid**
- **sntp reliable authentication-keyid**

# 目 录

1 PoE .....	1-1
1.1 PoE配置命令 .....	1-1
1.1.1 apply poe-profile .....	1-1
1.1.2 apply poe-profile interface .....	1-1
1.1.3 display poe device .....	1-2
1.1.4 display poe interface .....	1-3
1.1.5 display poe interface power .....	1-7
1.1.6 display poe pse .....	1-8
1.1.7 display poe pse interface .....	1-10
1.1.8 display poe pse interface power .....	1-11
1.1.9 display poe-profile .....	1-13
1.1.10 display poe-profile interface .....	1-14
1.1.11 poe detection-mode .....	1-15
1.1.12 poe enable .....	1-15
1.1.13 poe legacy enable .....	1-16
1.1.14 poe max-power .....	1-17
1.1.15 poe pd-description .....	1-18
1.1.16 poe pd-policy priority .....	1-18
1.1.17 poe priority .....	1-19
1.1.18 poe reset enable .....	1-20
1.1.19 poe update .....	1-20
1.1.20 poe utilization-threshold .....	1-21
1.1.21 poe-profile .....	1-22

# 1 PoE

## 1.1 PoE配置命令

### 1.1.1 apply poe-profile

**apply poe-profile** 命令用来将 PoE profile 应用到 PoE 接口。

**undo apply poe-profile** 命令用来恢复缺省情况。

#### 【命令】

```
apply poe-profile { index index | name profile-name }  
undo apply poe-profile { index index | name profile-name }
```

#### 【缺省情况】

未将 PoE profile 应用到 PoE 接口。

#### 【视图】

PoE 接口视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**index** *index*: PoE profile 的索引，取值范围为 1~100。

**name** *profile-name*: PoE profile 的名称，为 1~15 个字符的字符串，区分大小写。

#### 【举例】

# 将名为 forIPphone 的 PoE profile 应用到 PoE 接口 GigabitEthernet1/0/1。

```
<Sysname> system-view  
[Sysname] interface gigabitethernet 1/0/1  
[Sysname-GigabitEthernet1/0/1] apply poe-profile name forIPphone
```

#### 【相关命令】

- **apply poe-profile interface**
- **display poe-profile**

### 1.1.2 apply poe-profile interface

**apply poe-profile interface** 命令用来将 PoE profile 应用到 PoE 接口。

**undo apply poe-profile interface** 命令用来取消 PoE 接口上应用的 PoE profile。

#### 【命令】

```
apply poe-profile { index index | name profile-name } interface  
interface-range
```

```
undo apply poe-profile { index index | name profile-name } interface
interface-range
```

#### 【缺省情况】

未将 PoE profile 应用到 PoE 接口。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**index index**: PoE profile 的索引，取值范围为 1~100。

**name profile-name**: PoE profile 的名称，为 1~15 个字符，区分大小写。

**interface-range**: 以太网接口范围，表示多个以太网接口。表示方式为 *interface-range = interface-type interface-number [ to interface-type interface-number ]*。其中，*interface-type interface-number* 为接口类型和接口编号。起始接口号要小于结束接口号，结束接口要和起始接口是同种类型。接口范围可以任意，如果指定范围内存在不支持 PoE 的接口，PoE 配置文件应用时将忽略这类接口。

#### 【举例】

# 将名称为 forIPphone 的 PoE profile 应用到 PoE 接口 GigabitEthernet1/0/1。

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

```
[Sysname] apply poe-profile name forIPphone interface gigabitethernet 1/0/1
```

# 将索引为 1 的 PoE profile 应用到 PoE 接口 GigabitEthernet1/0/2 至 GigabitEthernet1/0/6。

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

```
[Sysname] apply poe-profile index 1 interface gigabitethernet 1/0/2 to gigabitethernet 1/0/6
```

#### 【相关命令】

- **apply poe-profile**
- **display poe-profile interface**

### 1.1.3 display poe device

**display poe device** 命令用来显示 PSE 的概要信息。

#### 【命令】

```
display poe device [ slot slot-number ]
```

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

network-operator

【参数】

**slot slot-number:** 显示成员设备上所有 PSE 的概要信息。*slot-number* 表示设备在 IRF 中的成员编号。不指定该参数时，表示所有成员设备。

【举例】

```
# 显示 PSE 的概要信息。
<Sysname> display poe device
Slot 1:
PSE ID    Slot No.  SSlot No.  PortNum    MaxPower(W)  State  Model
4         1        0          48         600.0        Off    LSPPSE48A
```

表1-1 display poe device 命令显示信息描述表

字段	描述
PSE ID	PSE编号
Slot No.	PSE所在槽位号
SSlot No.	PSE所在子槽位号
PortNum	PSE上PoE接口的数量
MaxPower(W)	PSE最大供电功率（单位为：瓦）
State	PSE状态： <ul style="list-style-type: none"><li>On: PSE 正在供电</li><li>Off: PSE 停止供电</li><li>Faulty: PSE 故障</li></ul>
Model	PSE型号

1.1.4 display poe interface

**display poe interface** 命令用来显示设备 PoE 接口的供电状态。

【命令】

```
display poe interface [ interface-type interface-number ]
```

【视图】

任意视图

【缺省用户角色】

```
network-admin
network-operator
```

【参数】

**interface-type interface-number:** 指定接口类型及接口编号，显示 PoE 接口的供电状态。如果未指定本参数，则显示所有 PoE 接口的供电状态。

【举例】

# 显示 PoE 接口 GigabitEthernet1/0/1 的供电状态。

```
<Sysname> display poe interface gigabitethernet 1/0/1
```

```
PoE Status           : Enabled
Power Priority        : Critical
Oper                 : On
IEEE Class           : 1
Detection Status     : Delivering power
Power Mode           : Signal
Current Power        : 11592    mW
Average Power        : 11610    mW
Peak Power           : 11684    mW
Max Power            : 15400    mW
Electric Current     : 244      mA
Voltage              : 51.7     V
PD Description       : IP Phone For Room 101
```

表1-2 display poe interface 命令显示信息描述表

字段	描述
PoE Status	PoE接口远程供电功能是否开启： <ul style="list-style-type: none"><li>• Enabled: 开启</li><li>• Disabled: 关闭</li></ul>
Power Priority	PoE接口供电优先级： <ul style="list-style-type: none"><li>• Critical: 最高</li><li>• High: 高</li><li>• Low: 低</li></ul>
Oper	PoE接口工作状态： <ul style="list-style-type: none"><li>• Off: 供电功能处于关闭状态</li><li>• On: 正在正常供电</li><li>• Power-lack: PSE 剩余保证功率不够，导致无法对优先级为 Critical 的 PoE 接口供电</li><li>• Power-deny: 拒绝供电，PD 要求功率大于配置功率</li><li>• Power-itself: 外接设备正在自己供电</li><li>• Power-limit: 正在受限供电，PD 要求功率大于配置功率，PSE 仍按配置功率供电</li></ul>
IEEE Class	由IEEE规定的PD功率等级，取值为：0、1、2、3、4 显示为“-”表示不支持

字段	描述
Detection Status	PoE接口检测状态： <ul style="list-style-type: none"> <li>Disabled: PoE 接口供电功能处于关闭状态</li> <li>Searching: 正在搜索 PD</li> <li>Delivering power: 正在向 PD 供电</li> <li>Fault: 错误</li> <li>Test: 测试状态</li> <li>Other fault: 其他错误状态</li> <li>PD disconnected: PD 未连接</li> </ul>
Power Mode	PoE接口供电方式： <ul style="list-style-type: none"> <li>Signal: 信号线供电方式</li> <li>Spare: 空闲线供电方式</li> </ul>
Current Power	PoE接口当前功率，包括PD消耗功率和传输损耗
Average Power	PoE接口平均功率
Peak Power	PoE接口峰值功率
Max Power	PoE接口最大功率
Electric Current	PoE接口当前电流
Voltage	PoE接口当前电压
PD Description	PoE接口所连接PD的描述信息，用于辅助用户识别PD的类型和位置等

# 显示设备所有 PoE 接口的供电状态。

```
<Sysname> display poe interface
```

Interface	PoE	Priority	CurPower (W)	Oper	IEEE Class	Detection Status
GE1/0/1	Disabled	Low	0.0	Off	0	Disabled
GE1/0/2	Enabled	Low	0.0	Off	0	Searching
GE1/0/3	Disabled	Low	0.0	Off	0	Disabled
GE1/0/4	Disabled	Low	0.0	Off	0	Disabled
GE1/0/5	Disabled	Low	0.0	Off	0	Disabled
GE1/0/6	Disabled	Low	0.0	Off	0	Disabled
GE1/0/7	Disabled	Low	0.0	Off	0	Disabled
GE1/0/8	Disabled	Low	0.0	Off	0	Disabled
GE1/0/9	Disabled	Low	0.0	Off	0	Disabled
GE1/0/10	Disabled	Low	0.0	Off	0	Disabled
GE1/0/11	Disabled	Low	0.0	Off	0	Disabled
GE1/0/12	Disabled	Low	0.0	Off	0	Disabled
GE1/0/13	Disabled	Low	0.0	Off	0	Disabled
GE1/0/14	Disabled	Low	0.0	Off	0	Disabled
GE1/0/15	Disabled	Low	0.0	Off	0	Disabled
GE1/0/16	Disabled	Low	0.0	Off	0	Disabled
GE1/0/17	Disabled	Low	0.0	Off	0	Disabled
GE1/0/18	Disabled	Low	0.0	Off	0	Disabled



```

GE1/0/19    Disabled Low    0.0    Off    0    Disabled
GE1/0/20    Disabled Low    0.0    Off    0    Disabled
GE1/0/21    Disabled Low    0.0    Off    0    Disabled
GE1/0/22    Disabled Low    0.0    Off    0    Disabled
GE1/0/23    Disabled Low    0.0    Off    0    Disabled
GE1/0/24    Disabled Low    0.0    Off    0    Disabled
    --- On State Ports: 0; Used: 0.0(W); Remaining: 370.0(W) ---

```

表1-3 display poe interface 命令显示信息描述表

字段	描述
Interface	PoE接口名称简称
PoE	PoE接口远程供电功能是否开启： <ul style="list-style-type: none"> <li>Enabled: 开启</li> <li>Disabled: 关闭</li> </ul>
Priority	PoE接口供电优先级： <ul style="list-style-type: none"> <li>Critical: 最高</li> <li>High: 高</li> <li>Low: 低</li> </ul>
CurPower	PoE接口功率
Oper	PoE接口工作状态： <ul style="list-style-type: none"> <li>Off: 供电功能处于关闭状态</li> <li>On: 正在正常供电</li> <li>Power-lack: 剩余保证功率不够，导致无法给 Critical 接口供电</li> <li>Power-deny: 拒绝供电，PD 要求功率大于配置功率</li> <li>Power-itself: 外接设备正在自己供电</li> <li>Power-limit: 正在受限供电，PD 要求功率大于配置功率，PSE 仍按配置功率供电</li> </ul>
IEEE Class	由IEEE规定的PD功率等级，取值为：0、1、2、3、4 显示为“-”表示不支持
Detection Status	PoE接口检测状态： <ul style="list-style-type: none"> <li>Disabled: PoE 接口供电处于关闭状态</li> <li>Searching: 正在搜索 PD</li> <li>Delivering Power: 正在向 PD 供电</li> <li>Fault: 错误</li> <li>Test: 测试状态</li> <li>Other Fault: 其他错误状态</li> <li>PD Disconnected: PD 未连接</li> </ul>
On State Ports	正在供电的PoE接口数量
Used	当前供电PoE接口消耗的功率
Remaining	系统总剩余功率

## 1.1.5 display poe interface power

**display poe interface power** 命令用来显示 PoE 接口的功率信息。

### 【命令】

**display poe interface power** [ *interface-type interface-number* ]

### 【视图】

任意视图

### 【缺省用户角色】

network-admin

network-operator

### 【参数】

*interface-type interface-number*: 指定接口类型及接口编号, 显示 PoE 接口的功率信息。  
如果未指定本参数, 则显示所有 PoE 接口的功率信息。

### 【举例】

# 显示 PoE 接口 GigabitEthernet1/0/1 的功率信息。

```
<Sysname> display poe interface power gigabitethernet 1/0/1
```

Interface	Current (W)	Peak (W)	Max (W)	PD Description
GE1/0/1	15.0	15.3	30.0	Access Point on Room 509 for Peter

# 显示所有 PoE 接口的功率信息。

```
<Sysname> display poe interface power
```

Interface	Current (W)	Peak (W)	Max (W)	PD Description
GE1/0/1	0.0	0.0	30.0	
GE1/0/2	0.0	0.0	30.0	
GE1/0/3	0.0	0.0	30.0	
GE1/0/4	0.0	0.0	30.0	
GE1/0/5	0.0	0.0	30.0	
GE1/0/6	0.0	0.0	30.0	
GE1/0/7	0.0	0.0	30.0	
GE1/0/8	0.0	0.0	30.0	
GE1/0/9	0.0	0.0	30.0	
GE1/0/10	0.0	0.0	30.0	
GE1/0/11	0.0	0.0	30.0	
GE1/0/12	0.0	0.0	30.0	
GE1/0/13	0.0	0.0	30.0	
GE1/0/14	0.0	0.0	30.0	
GE1/0/15	0.0	0.0	30.0	
GE1/0/16	0.0	0.0	30.0	
GE1/0/17	0.0	0.0	30.0	
GE1/0/18	0.0	0.0	30.0	
GE1/0/19	0.0	0.0	30.0	
GE1/0/20	0.0	0.0	30.0	

```

GE1/0/21      0.0      0.0      30.0
GE1/0/22      0.0      0.0      30.0
GE1/0/23      0.0      0.0      30.0
GE1/0/24      0.0      0.0      30.0
--- On State Ports: 0; Used: 0.0(W); Remaining: 370.0(W) ---

```

表1-4 display poe interface power 命令显示信息描述表

字段	描述
Interface	PoE接口简称
CurPower	PoE接口当前功率
PeakPower	PoE接口峰值功率
MaxPower	PoE接口最大功率
PD Description	PoE接口连接PD描述信息，用于辅助用户识别PD的类型和位置等
Ports On	正在供电的PoE接口数量
Used	所有PoE接口当前消耗功率
Remaining	系统总剩余功率

## 1.1.6 display poe pse

**display poe pse** 命令用来显示 PSE 的详细信息。

### 【命令】

```
display poe pse [ pse-id ]
```

### 【视图】

任意视图

### 【缺省用户角色】

```
network-admin
network-operator
```

### 【参数】

*pse-id*: PSE 编号。如果不指定该参数，则显示所有的 PSE 的详细信息。

### 【举例】

# 显示设备 PSE 的详细信息。

```

<Sysname> display poe pse
PSE ID           : 4
Slot No.         : 1
SSlot No.        : 0
PSE Model        : LSPPSE48A
PSE Status       : Enabled
Power Priority    : Low
Current Power     : 0.0    W

```

```

Average Power          : 0.0      W
Peak Power             : 0.0      W
Max Power              : 600.0    W
Remaining Guaranteed Power : 600.0    W
PSE CPLD Version       : -
PSE Software Version   : 172
PSE Hardware Version   : 0
Legacy PD Detection    : Disabled
Power Utilization Threshold : 80
PD Power Policy        : Priority
PD Disconnect-Detection Mode : DC

```

表1-5 display poe pse 命令显示信息描述表

字段	描述
PSE ID	PSE编号
SSlot No.	PSE所在子槽位号
PSE Model	PSE模块型号
PSE Status	PSE供电状态: <ul style="list-style-type: none"> <li>Enabled: 已使能</li> <li>Disabled: 未使能</li> </ul>
Power Priority	PSE供电优先级
Current Power	PSE当前功率
Average Power	PSE平均功率
Peak Power	PSE峰值功率
Max Power	PSE最大功率
Remaining Guaranteed Power	PSE剩余保证功率=PSE最大保证功率-该PSE中优先级为Critical的接口最大功率之和
PSE CPLD Version	PSE CPLD (Complex Programmable Logical Device, 复杂可编程逻辑器件) 版本
PSE Software Version	PSE软件版本
PSE Hardware Version	PSE硬件版本
Legacy PD Detection	PSE非标准PD检测: <ul style="list-style-type: none"> <li>Enabled: 开启</li> <li>Disabled: 关闭</li> </ul>
Power Utilization Threshold	PSE功率告警阈值
PD Power Policy	PD功率管理策略模式
PD Disconnect-Detection Mode	PD断开检测方式

### 1.1.7 display poe pse interface

**display poe pse interface** 命令用来显示 PSE 上 PoE 接口的供电状态。

#### 【命令】

**display poe pse *pse-id* interface**

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

network-operator

#### 【参数】

**pse *pse-id***: PSE 编号，可以用 **display poe device** 命令查看 PSE 编号和槽号的对应关系。

#### 【举例】

# 显示 PSE 4 上连接的所有 PoE 接口的供电状态。

```
<Sysname> display poe pse 4 interface
Interface    PoE      Priority  CurPower  Oper    IEEE  Detection
              (W)
              Class Status
GE1/0/1      Disabled Low       0.0      Off     0      Disabled
GE1/0/2      Enabled  Low       0.0      Off     0      Searching
GE1/0/3      Disabled Low       0.0      Off     0      Disabled
GE1/0/4      Disabled Low       0.0      Off     0      Disabled
GE1/0/5      Disabled Low       0.0      Off     0      Disabled
GE1/0/6      Disabled Low       0.0      Off     0      Disabled
GE1/0/7      Disabled Low       0.0      Off     0      Disabled
GE1/0/8      Disabled Low       0.0      Off     0      Disabled
GE1/0/9      Disabled Low       0.0      Off     0      Disabled
GE1/0/10     Disabled Low       0.0      Off     0      Disabled
GE1/0/11     Disabled Low       0.0      Off     0      Disabled
GE1/0/12     Disabled Low       0.0      Off     0      Disabled
GE1/0/13     Disabled Low       0.0      Off     0      Disabled
GE1/0/14     Disabled Low       0.0      Off     0      Disabled
GE1/0/15     Disabled Low       0.0      Off     0      Disabled
GE1/0/16     Disabled Low       0.0      Off     0      Disabled
GE1/0/17     Disabled Low       0.0      Off     0      Disabled
GE1/0/18     Disabled Low       0.0      Off     0      Disabled
GE1/0/19     Disabled Low       0.0      Off     0      Disabled
GE1/0/20     Disabled Low       0.0      Off     0      Disabled
GE1/0/21     Disabled Low       0.0      Off     0      Disabled
GE1/0/22     Disabled Low       0.0      Off     0      Disabled
GE1/0/23     Disabled Low       0.0      Off     0      Disabled
GE1/0/24     Disabled Low       0.0      Off     0      Disabled
--- On State Ports: 0; Used: 0.0(W); Remaining: 370.0(W) ---
```

表1-6 display poe pse interface 命令显示信息描述表

字段	描述
Interface	PoE接口简称
PoE	PoE接口远程供电功能是否开启： <ul style="list-style-type: none"> <li>Enabled: 开启</li> <li>Disabled: 关闭</li> </ul>
Priority	PoE接口供电优先级： <ul style="list-style-type: none"> <li>Critical: 最高</li> <li>High: 高</li> <li>Low: 低</li> </ul>
CurPower	PoE接口当前功率
Oper	PoE接口工作状态： <ul style="list-style-type: none"> <li>Off: 供电功能处于关闭状态</li> <li>On: 正在正常供电</li> <li>Power-lack: 剩余保证功率不够，导致无法给 Critical 接口供电</li> <li>Power-deny: 拒绝供电，PD 要求功率大于配置功率</li> <li>Power-itself: 外接设备正在自己供电</li> <li>Power-limit: 正在受限供电，PD 要求功率大于配置功率，PSE 仍按配置功率供电</li> </ul>
IEEE Class	PD功率等级
Detection Status	PoE接口检测状态： <ul style="list-style-type: none"> <li>Disabled: PoE 接口供电处于关闭状态</li> <li>Searching: 正在搜索 PD</li> <li>Delivering Power: 正在向 PD 供电</li> <li>Fault: 错误</li> <li>Test: 测试状态</li> <li>Other Fault: 其他错误状态</li> <li>PD Disconnected: PD 未连接</li> </ul>
On State Ports	正在供电的PoE接口数量
Used	该PSE上供电PoE接口消耗的功率
Remaining	此PSE上剩余功率

### 1.1.8 display poe pse interface power

**display poe pse interface power** 命令用来显示 PSE 上 PoE 接口的功率信息。

#### 【命令】

**display poe pse *pse-id* interface power**

【视图】

任意视图

【缺省用户角色】

network-admin  
network-operator

【参数】

pse pse-id: PSE 编号，可以用 **display poe device** 命令查看 PSE 编号和槽号的对应关系。

【举例】

```
# 显示 PSE 4 连接的 PoE 接口的功率信息。
<Sysname> display poe pse 4 interface power
Interface      Current      Peak          Max          PD Description
              (W)          (W)           (W)
GE1/0/1        0.0          0.0           30.0
GE1/0/2        0.0          0.0           30.0
GE1/0/3        0.0          0.0           30.0
GE1/0/4        0.0          0.0           30.0
GE1/0/5        0.0          0.0           30.0
GE1/0/6        0.0          0.0           30.0
GE1/0/7        0.0          0.0           30.0
GE1/0/8        0.0          0.0           30.0
GE1/0/9        0.0          0.0           30.0
GE1/0/10       0.0          0.0           30.0
GE1/0/11       0.0          0.0           30.0
GE1/0/12       0.0          0.0           30.0
GE1/0/13       0.0          0.0           30.0
GE1/0/14       0.0          0.0           30.0
GE1/0/15       0.0          0.0           30.0
GE1/0/16       0.0          0.0           30.0
GE1/0/17       0.0          0.0           30.0
GE1/0/18       0.0          0.0           30.0
GE1/0/19       0.0          0.0           30.0
GE1/0/20       0.0          0.0           30.0
GE1/0/21       0.0          0.0           30.0
GE1/0/22       0.0          0.0           30.0
GE1/0/23       0.0          0.0           30.0
GE1/0/24       0.0          0.0           30.0
--- On State Ports: 0; Used: 0.0(W); Remaining: 370.0(W) ---
```

表1-7 display poe pse interface power 命令显示信息描述表

字段	描述
Interface	PoE接口简称
Current	PoE接口当前功率
Peak	PoE接口峰值功率

字段	描述
Max	PoE接口最大功率
PD Description	PoE接口连接PD描述信息，用于辅助用户识别PD的类型和位置等
Ports On	正在供电的PoE接口数量
Used	所有PoE接口当前消耗功率
Remaining	此PSE上的剩余功率

### 1.1.9 display poe-profile

**display poe-profile** 命令用来显示 PoE Profile 的相关信息。

#### 【命令】

**display poe-profile** [ **index** *index* | **name** *profile-name* ]

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

**index** *index*: PoE Profile 的索引，取值范围为 1~100。

**name** *profile-name*: PoE Profile 的名称，为 1~15 个字符的字符串，区分大小写。

#### 【使用指导】

如果不指定参数，将显示已存在的所有的 PoE Profile 的配置和应用信息。

#### 【举例】

# 显示所有 PoE Profile 的相关信息。

```
<Sysname> display poe-profile
PoE Profile      Index  ApplyNum  Interfaces  Configuration
forIPhone        1       4        GE1/0/1     poe enable
                  GE1/0/2     poe priority critical
                  GE1/0/3
                  GE1/0/4
forAP             2       2        GE1/0/5     poe enable
                  GE1/0/6     poe max-power 14000
--- Total PoE profiles: 2, total ports: 6 ---
```

# 显示索引为 1 的 PoE Profile 的相关信息。

```
<Sysname> display poe-profile index 1
PoE Profile      Index  ApplyNum  Interfaces  Configuration
forIPhone        1       6        GE1/0/1     poe enable
                  GE1/0/2     poe priority critical
                  GE1/0/3
```



GE1/0/4  
GE1/0/5  
GE1/0/6

--- Total ports: 6 ---

表1-8 display poe-profile 命令显示信息描述表

字段	描述
PoE Profile	PoE Profile的名称
Index	PoE Profile的索引
ApplyNum	应用到的PoE接口数量
Interfaces	应用了PoE Profile的PoE接口名称简称
Configuration	PoE Profile的配置项
Total PoE profiles	创建的PoE Profile数目
total ports	应用PoE Profile的PoE接口数目

### 1.1.10 display poe-profile interface

**display poe-profile interface** 命令用来显示 PoE 接口生效的 PoE Profile 配置项和应用的信息。

#### 【命令】

**display poe-profile interface** *interface-type interface-number*

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

*interface-type interface-number*: 指定接口类型及接口编号。

#### 【举例】

# 显示 PoE 接口 GigabitEthernet1/0/1 的 PoE Profile 配置和应用的所有信息。

```
<Sysname> display poe-profile interface gigabitethernet 1/0/1
PoEProfile      Index  ApplyNum  Interface  Effective configuration
forIPhone       1      6         GE1/0/1    poe enable
                                   poe priority critical
```

因为PoE Profile的配置项（Configuration）可能只有部分应用成功，所以显示的是该接口生效的配置项（Effective configuration），其他字段的描述请参见 [表 1-8](#)。

### 1.1.11 poe detection-mode

**poe detection-mode** 命令用来配置对 PD 支持标准的检测方式。

**undo poe detection-mode** 命令用来恢复缺省情况。

#### 【命令】

```
poe detection-mode { none | simple | strict }  
undo poe detection-mode
```

#### 【缺省情况】

对 PD 支持标准的检测方式为 **strict**。

#### 【视图】

PoE 接口视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**none**: 不检测，只要与 PD 设备连接无短路和断路，PoE 接口即可对 PD 供电。

**simple**: 简单检测方式，PD 满足 IEEE 802.af 或 802.at 最低要求即可对 PD 供电。

**strict**: 严格检测方式，PD 需要满足 IEEE 802.af 或 802.at 所有要求时，才会对 PD 供电。

#### 【使用指导】

PoE 接口连接非 PD 设备时，请勿配置 **none** 参数，否则，设备会尝试为非 PD 设备供电，可能造成非 PD 设备元器件损坏。

使用该命令可以检测标准和非标准的 PD。如需检测非标准的 PD，请先配置 **poe legacy enable** 命令。

仅 Release 6127 及以上版本支持本命令。

#### 【举例】

# 配置 GigabitEthernet1/0/1 端口对 PD 支持标准的检测方式为简单检测方式。

```
<Sysname> system-view  
[Sysname] interface gigabitethernet 1/0/1  
[Sysname-GigabitEthernet1/0/1] poe detection-mode simple
```

#### 【相关命令】

- **poe legacy enable**

### 1.1.12 poe enable

**poe enable** 命令用来开启 PoE 接口远程供电功能。

**undo poe enable** 命令用来关闭 PoE 接口远程供电功能。

#### 【命令】

```
poe enable  
undo poe enable
```

### 【缺省情况】

对于 R6126P12 版本，PoE 接口远程供电功能处于关闭状态。

对于 R6126P20 及以上版本：

- 设备采用出厂配置启动时，PoE 接口远程供电功能处于开启状态。
- 设备采用空配置启动时，PoE 接口远程供电功能处于关闭状态。

关于空配置启动和缺省配置启动的详细介绍，请参见“基础配置指导”中的“配置文件管理”。

### 【视图】

PoE 接口视图

PoE profile 视图

### 【缺省用户角色】

network-admin

### 【使用指导】

在 PoE profile 视图下配置时，如果该 PoE profile 已经应用到 PoE 接口，需先取消该 PoE profile 在 PoE 接口的应用。

在接口视图下配置时，若已用 PoE 配置文件对该 PoE 接口进行过配置，应先取消 PoE 配置文件在该 PoE 接口的应用。

### 【举例】

# 开启 PoE 接口远程供电功能。

```
<Sysname> system-view
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] poe enable
```

#在 PoE profile abc 中，开启 PoE 接口远程供电功能。

```
<Sysname> system-view
[Sysname] poe-profile abc
[Sysname-poe-profile-abc-1] poe enable
```

### 【相关命令】

- **display poe interface**
- **poe-profile**

#### 1.1.13 poe legacy enable

**poe legacy enable** 命令用来开启 PSE 的检测非标准 PD 功能。

**undo poe legacy enable** 命令用来关闭 PSE 的检测非标准 PD 功能。

### 【命令】

```
poe legacy enable pse pse-id
undo poe legacy enable pse pse-id
```

### 【缺省情况】

PSE 检测非标准 PD 功能处于关闭状态。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**pse** *pse-id*: PSE 编号。

### 【举例】

# 开启 PSE 4 的检测非标准 PD 功能。

```
<Sysname> system-view
[Sysname] poe legacy enable pse 4
```

### 【相关命令】

- **display poe pse**

## 1.1.14 poe max-power

**poe max-power** 命令用来配置 PoE 接口的最大功率。

**undo poe max-power** 命令用来恢复缺省情况。

### 【命令】

```
poe max-power max-power
undo poe max-power
```

### 【缺省情况】

PoE 接口的最大功率为 30000。

### 【视图】

PoE 接口视图

PoE profile 视图

### 【缺省用户角色】

network-admin

### 【参数】

**max-power**: 为 PoE 接口分配的最大供电功率，单位为毫瓦，按照一定的步长取值。取值范围为 1000~30000。

### 【举例】

# 配置 PoE 接口的最大功率为 12000 毫瓦。

```
<Sysname> system-view
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] poe max-power 12000
```

# 通过 PoE profile 配置 PoE 接口的最大供电功率为 12000 毫瓦。

```
<Sysname> system-view
[Sysname] poe-profile abc
```

```
[Sysname-poe-profile-abc-1] poe max-power 12000
```

### 1.1.15 poe pd-description

**poe pd-description** 命令用来配置 PoE 接口连接 PD 的描述信息。

**undo poe pd-description** 命令用来恢复缺省情况。

#### 【命令】

```
poe pd-description text
undo poe pd-description
```

#### 【缺省情况】

未配置 PoE 接口连接 PD 的描述信息。

#### 【视图】

PoE 接口视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**text**: PoE 接口连接 PD 的描述信息，为 1~80 个字符的字符串，区分大小写。

#### 【举例】

# 配置 PoE 接口连接 PD 的描述信息为连接 101 室的 IP 电话。

```
<Sysname> system-view
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] poe pd-description IP Phone For Room 101
```

### 1.1.16 poe pd-policy priority

**poe pd-policy priority** 命令用来开启 PoE 接口功率管理优先级策略。

**undo poe pd-policy priority** 命令用来恢复缺省情况。

#### 【命令】

```
poe pd-policy priority
undo poe pd-policy priority
```

#### 【缺省情况】

未配置 PoE 接口功率管理优先级策略。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【使用指导】

在没有开启 PoE 接口功率管理优先级策略时，如果 PSE 功率过载，则不对新接入的 PD 供电。

在开启 PoE 接口功率管理优先级策略时，如果 PSE 功率过载，接入新的 PD，将对优先级低的 PD 断电，保证优先级高的 PD 供电。

#### 【举例】

# 配置 PD 功率管理策略为优先级策略。

```
<Sysname> system-view
[Sysname] poe pd-policy priority
```

#### 【相关命令】

- **poe priority**

### 1.1.17 poe priority

**poe priority** 命令用来配置 PoE 接口供电优先级。

**undo poe priority** 命令用来恢复缺省情况。

#### 【命令】

```
poe priority { critical | high | low }
undo poe priority
```

#### 【缺省情况】

PoE 接口供电优先级为 **low**。

#### 【视图】

PoE 接口视图

PoE profile 视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**critical**: 配置 PoE 接口供电优先级为最高，即将该 PoE 接口置为供电保证模式，插入该接口的 PD 可以以最高优先级得到供电。

**high**: 配置 PoE 接口供电优先级为高。

**low**: 配置 PoE 接口供电优先级为低。

#### 【使用指导】

当 PSE 功率过载的情况下，优先对供电优先级高的 PoE 接口进行供电。

在 PoE 配置文件视图下配置时，如果该 PoE 配置文件已经应用到 PoE 接口，需先取消该 PoE 配置文件在 PoE 接口的应用。

在 PoE 接口视图下配置时，若已用 PoE 配置文件对该 PoE 接口进行过配置，应先取消 PoE 配置文件在该 PoE 接口的应用。

如果配置了相同的优先级，接口编号小的 PoE 接口的优先级高。

#### 【举例】

# 配置 PoE 接口供电优先级为 Critical。

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

```
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] poe priority critical
# 通过 PoE profile 配置 PoE 接口供电优先级为 Critical。
<Sysname> system-view
[Sysname] poe-profile abc
[Sysname-poe-profile-abc-1] poe priority critical
[Sysname-poe-profile-abc-1] quit
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] apply poe-profile name abc
```

#### 【相关命令】

- **poe pd-policy priority**

### 1.1.18 poe reset enable

**poe reset enable** 命令用来开启设备热重启后主动复位 PoE 端口供电功能。

**undo poe reset enable** 命令用来关闭设备热重启后主动复位 PoE 端口供电功能。

#### 【命令】

```
poe reset enable
undo poe reset enable
```

#### 【缺省情况】

设备热重启后主动复位 PoE 端口供电功能处于关闭状态。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【使用指导】

设备热重启（通过执行 **reboot** 命令重启）过程中 PoE 端口将继续向 PD 供电，但此时设备与 PD 的数据连接已中断，设备重启完成后，PD 可能无法再次主动与设备建立数据连接。配置该功能后，设备热重启时主动对 PoE 端口断电并再次供电，此时 PD 将与设备重新建立数据连接。

#### 【举例】

# 开启设备热重启后主动复位 PoE 端口供电功能。

```
<Sysname> system-view
[Sysname] poe reset enable
```

### 1.1.19 poe update

**poe update** 命令用来在线升级 PSE 固件。

#### 【命令】

```
poe update { full | refresh } filename [ pse pse-id ]
```

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**full**: 用 **full** 模式升级 PSE 固件。

**refresh**: 用 **refresh** 模式升级 PSE 固件。

**filename**: 升级文件的名称，为 1~64 个字符的字符串，不区分大小写。该文件必须在设备文件系统的根目录下。

**pse** *pse-id*: PSE 编号，不指定该参数表示升级所有 PSE 固件。

## 【使用指导】

**refresh** 模式是在原 PSE 固件的基础上进行更新升级，升级过程简单快速。一般情况下使用该模式来升级 PSE 固件。

**full** 模式是将原 PSE 固件删除，再安装新 PSE 固件。PSE 固件被损坏的情况下（表现为所有的 PoE 命令执行不成功），可用 **full** 模式进行升级，使软件恢复。

## 【举例】

# 在线升级 PSE 4 固件。

```
<Sysname> system-view  
[Sysname] poe update refresh POE-168.bin pse 4
```

### 1.1.20 poe utilization-threshold

**poe utilization-threshold** 命令用来配置 PSE 的功率告警阈值。

**undo poe utilization-threshold** 命令用来恢复缺省情况。

## 【命令】

```
poe utilization-threshold value pse pse-id  
undo poe utilization-threshold pse pse-id
```

## 【缺省情况】

PSE 的功率告警阈值为 80%。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**value**: 功率告警阈值,取值范围为 1~99，单位为百分比。

**pse** *pse-id*: PSE 编号。



### 【使用指导】

当 PSE 的功率使用百分比首次超过或者低于设置的告警阈值时，系统将生成告警信息，发送给设备的 SNMP 模块，通过设置 SNMP 中告警信息的发送参数，来决定告警信息输出的相关属性。有关告警信息的详细介绍，请参见“网络管理和监控配置指导”中的“SNMP”。

### 【举例】

# 配置 PSE 4 的功率告警阈值为 90%。

```
<Sysname> system-view
[Sysname] poe utilization-threshold 90 pse 4
```

## 1.1.21 poe-profile

**poe-profile** 命令用来创建 PoE profile，并进入 PoE profile 视图。如果指定的 PoE profile 已经存在，则直接进入 PoE profile。

**undo poe-profile** 命令用来删除 PoE profile。

### 【命令】

```
poe-profile profile-name [index]
undo poe-profile { index index | name profile-name }
```

### 【缺省情况】

不存在 PoE profile。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**profile-name**: PoE profile 的名称，为 1~15 个字符的字符串，区分大小写。以英文字母[a-z,A-Z]开始，并且不能为保留关键字 **undo**、**all**、**name**、**interface**、**user**、**poe**、**disable**、**max-power**、**mode**、**priority** 和 **enable** 等。

**index**: PoE profile 的索引，取值范围为 1~100。

### 【使用指导】

批量配置 PoE 接口时，一般采用 PoE Profile 配置。如果不指定索引值，系统会为此 PoE profile 自动分配索引，从 1 开始。

如果 PoE profile 已经应用到接口，不允许删除该 PoE profile，必须先执行 **undo apply poe-profile**，取消 PoE profile 在 PoE 接口的应用后，才能删除该 PoE profile。

### 【举例】

# 创建名称为 abc 的 PoE profile，指定索引为 3。

```
<Sysname> system-view
[Sysname] poe-profile abc 3
[Sysname-poe-profile-abc-3]
```

#创建名称为 def 的 PoE profile，不指定索引。

```
<Sysname> system-view  
[Sysname] poe-profile def  
[Sysname-poe-profile-def-1]
```

#### 【相关命令】

- **apply poe-profile**
- **poe enable**
- **poe max-power**
- **poe priority**

# 目 录

1 SNMP .....	1-1
1.1 SNMP配置命令 .....	1-1
1.1.1 display snmp-agent community.....	1-1
1.1.2 display snmp-agent context .....	1-3
1.1.3 display snmp-agent group .....	1-3
1.1.4 display snmp-agent local-engineid.....	1-4
1.1.5 display snmp-agent mib-node.....	1-5
1.1.6 display snmp-agent mib-view .....	1-9
1.1.7 display snmp-agent remote.....	1-11
1.1.8 display snmp-agent statistics.....	1-12
1.1.9 display snmp-agent sys-info.....	1-14
1.1.10 display snmp-agent trap queue .....	1-14
1.1.11 display snmp-agent trap-list.....	1-15
1.1.12 display snmp-agent usm-user .....	1-16
1.1.13 enable snmp trap updown.....	1-18
1.1.14 snmp-agent .....	1-18
1.1.15 snmp-agent { inform   trap } source .....	1-19
1.1.16 snmp-agent calculate-password .....	1-20
1.1.17 snmp-agent community .....	1-21
1.1.18 snmp-agent community-map .....	1-24
1.1.19 snmp-agent context.....	1-25
1.1.20 snmp-agent group.....	1-25
1.1.21 snmp-agent local-engineid .....	1-28
1.1.22 snmp-agent log.....	1-28
1.1.23 snmp-agent mib-view .....	1-29
1.1.24 snmp-agent packet max-size .....	1-31
1.1.25 snmp-agent packet response dscp.....	1-31
1.1.26 snmp-agent port .....	1-32
1.1.27 snmp-agent remote .....	1-33
1.1.28 snmp-agent sys-info contact.....	1-33
1.1.29 snmp-agent sys-info location .....	1-34
1.1.30 snmp-agent sys-info version.....	1-35
1.1.31 snmp-agent target-host .....	1-36

1.1.32 snmp-agent trap enable.....	1-38
1.1.33 snmp-agent trap if-mib link extended.....	1-39
1.1.34 snmp-agent trap life.....	1-39
1.1.35 snmp-agent trap log.....	1-40
1.1.36 snmp-agent trap queue-size .....	1-41
1.1.37 snmp-agent usm-user { v1   v2c } .....	1-41
1.1.38 snmp-agent usm-user v3 .....	1-43
1.1.39 snmp-agent usm-user v3 user-role.....	1-47

# 1 SNMP



## 说明

设备运行于 FIPS 模式时，本特性部分配置相对于非 FIPS 模式有所变化，具体差异请见本文相关描述。有关 FIPS 模式的详细介绍请参见“安全配置指导”中的“FIPS”。

## 1.1 SNMP配置命令

SNMP 告警信息包括 SNMP Trap 和 Inform 信息，用来告知 NMS 设备上发生了重要事件，比如，用户的登录/退出，接口状态变成 up/down 等。如无特殊说明，本文中的告警信息均指 Trap 和 Inform 两种信息。

### 1.1.1 display snmp-agent community

**display snmp-agent community** 命令用来显示 SNMPv1 或 SNMPv2c 的团体信息。

#### 【命令】

```
display snmp-agent community [ read | write ]
```

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

**read:** 显示只读访问权限的团体信息。

**write:** 显示读写访问权限的团体信息。

#### 【使用指导】

FIPS 模式下，不支持本命令。

不带参数时，显示所有 SNMP 团体的信息。

用户有两种方式创建团体：

- 使用 **snmp-agent community** 命令来创建团体。
- 配置 **snmp-agent usm-user { v1 | v2c }** 和 **snmp-agent group { v1 | v2c }** 命令成功创建 SNMPv1 或 SNMPv2c 用户以及相应的组后，系统会以用户名为团体名自动创建一个团体。

**display snmp-agent community** 会显示这两种方式创建的团体的信息。

【举例】

# 显示设备当前所有已配置的团体信息。

```
<Sysname> display snmp-agent community
Community name: aa
    Group name: aa
    ACL:2001
    Storage-type: nonVolatile
    Context name: con1

Community name: bb
    Role name: bb
    Storage-type: nonVolatile

Community name: userv1
    Group name: testv1
    Storage-type: nonVolatile

Community name: cc
    Group name: cc
    ACL name: testacl
    Storage-type: nonVolatile
```

表1-1 display snmp-agent community 命令显示信息描述表

字段	描述
Community name	团体名： <ul style="list-style-type: none"><li>如果团体是通过 <b>snmp-agent community</b> 命令创建的，则显示的是团体名</li><li>如果团体名是通过 <b>snmp-agent usm-user { v1   v2c }</b>命令创建的，则显示的是用户名</li></ul>
Group name	组名： <ul style="list-style-type: none"><li>如果团体名是通过 <b>snmp-agent community</b> 命令的 VACM 方式创建的，则组名和团体名相同</li><li>如果团体名是通过 <b>snmp-agent usm-user { v1   v2c }</b>命令创建的，则显示用户所在的组名</li></ul>
Role name	SNMP用户所在团体绑定的角色名： 通过 <b>snmp-agent community</b> 命令的RBAC方式创建的团体名可绑定用户角色
ACL	使用的ACL列表的编号（该字段仅在团体名与ACL绑定后显示，不会与ACL name同时存在）
ACL name	使用的ACL列表的名称（该字段仅在团体名与ACL名称绑定后显示，不会与ACL同时存在）
Storage-type	表示存储方式，分为以下几种： <ul style="list-style-type: none"><li><b>volatile</b>：重启后信息丢失</li><li><b>nonVolatile</b>：重启后信息仍保存</li><li><b>permanent</b>：重启后信息仍保存，允许更改，但不许删除</li><li><b>readOnly</b>：重启后信息仍保存，既不允许更改，也不许删除</li><li><b>other</b>：其他</li></ul>

字段	描述
Context name	SNMP上下文： <ul style="list-style-type: none"> <li>如果此团体名配置了对应的上下文映射，则显示对应的上下文</li> <li>如果此团体名未配置对应的上下文映射，该字段显示为空</li> </ul>

#### 【相关命令】

- `snmp-agent community`
- `snmp-agent usm-user { v1 | v2c }`

### 1.1.2 display snmp-agent context

`display snmp-agent context` 命令用来显示 SNMP 上下文。

#### 【命令】

`display snmp-agent context [ context-name ]`

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

*context-name*：显示指定的 SNMP 上下文，为 1~32 个字符的字符串，区分大小写。不指定该参数时，显示设备上所有已创建的 SNMP 上下文。

#### 【举例】

# 显示设备上所有已创建的 SNMP 上下文。

```
<Sysname> display snmp-agent context
testcontext
```

#### 【相关命令】

- `snmp-agent context`

### 1.1.3 display snmp-agent group

`display snmp-agent group` 命令用来显示 SNMP 组信息。

#### 【命令】

`display snmp-agent group [ group-name ]`

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

network-operator

【参数】

*group-name*: 非 FIPS 模式下，指定要显示信息的 SNMPv1、SNMPv2c 或 SNMPv3 组的名称；FIPS 模式下，指定要显示信息的 SNMPv3 组的名称。取值范围为 1~32 个字符的字符串，区分大小写。不指定该参数时，显示设备上所有已创建的 SNMP 组的信息。

【举例】

```
# 显示所有 SNMP 组的信息。
<Sysname> display snmp-agent group
  Group name: groupv3
    Security model: v3 noAuthnoPriv
    Readview: ViewDefault
    Writeview: <no specified>
    Notifyview: <no specified>
    Storage-type: nonVolatile
    ACL name: testacl
```

表1-2 display snmp-agent group 命令显示信息描述表

字段	描述
Group name	SNMP组名
Security model	SNMP组配置的安全模式，包括版本信息和安全模式，以空格分隔： <ul style="list-style-type: none"><li>对于 SNMPv1 和 SNMPv2c 版本，认证加密级别只能为 noAuthNoPriv（无认证无加密）</li><li>对于 SNMPv3 版本，安全模式分为三种：authPriv（既认证又加密）、authNoPriv（只认证不加密）、noAuthNoPriv（不认证不加密）</li></ul>
Readview	SNMP组对应的只读的MIB视图名
Writeview	SNMP组对应的可写的MIB视图名
Notifyview	SNMP组对应的可以发Trap和Inform信息的MIB视图名
Storage-type	存储方式，分为以下几种：volatile、nonVolatile、permanent、readOnly、other，具体描述请参见 <a href="#">表1-1</a>
ACL	使用的ACL列表的编号（该字段仅在SNMP组与ACL绑定后显示，不会与ACL name同时存在）
ACL name	使用的ACL列表的名称（该字段仅在SNMP组与ACL名称绑定后显示，不会与ACL同时存在）

【相关命令】

- snmp-agent group

1.1.4 display snmp-agent local-engineid

display snmp-agent local-engineid 命令用来显示本设备的 SNMP 引擎 ID。

【命令】

```
display snmp-agent local-engineid
```



### 【视图】

任意视图

### 【缺省用户角色】

network-admin  
network-operator

### 【使用指导】

SNMP 引擎 ID 是 SNMP 实体的唯一标识,它在一个 SNMP 管理域内是唯一的。SNMP 引擎是 SNMP 实体的重要组成部分,完成 SNMP 信息的信息调度、信息处理、安全验证、访问控制等功能。

### 【举例】

# 显示本设备的 SNMP 引擎 ID。

```
<Sysname> display snmp-agent local-engineid  
SNMP local engine ID: 800063A2800084E52BED7900000001
```

### 【相关命令】

- **snmp-agent local-engineid**

## 1.1.5 display snmp-agent mib-node

**display snmp-agent mib-node** 命令用来显示 SNMP 支持的 MIB 节点信息。

### 【命令】

```
display snmp-agent mib-node [ details | index-node | trap-node | verbose ]
```

### 【视图】

任意视图

### 【缺省用户角色】

network-admin  
network-operator

### 【参数】

**details:** 表示显示 SNMP 支持的 MIB 节点细节信息,包括节点名、OID 末位、下一个叶子节点名。

**index-node:** 显示 SNMP 支持的 MIB 表、节点名及索引节点 OID。

**trap-node:** 显示 SNMP 支持的 MIB 告警节点名及对应的 OID、告警绑定变量节点名及对应的 OID。

**verbose:** 显示 SNMP 支持的 MIB 节点详细信息,包括节点名、OID、节点类型、访问权限、数据类型,对应 MOR (Managed Object Repository, 管理对象库) 定义、父子兄弟节点信息等。

### 【使用指导】

未指定任何参数时,显示 SNMP 支持的 MIB 节点信息,包括节点名、OID 和节点访问权限。特性包中可以包含不同的 MIB 插件,设备根据加载特性包的不同,支持的 MIB 不相同。

【举例】

# 显示 SNMP 支持 MIB 节点信息。

```
<Sysname> display snmp-agent mib-node
```

```
iso<1>(NA)
  |-std<1.0>(NA)
    |-iso8802<1.0.8802>(NA)
      |-ieee802dot1<1.0.8802.1>(NA)
        |-ieee802dot1mibs<1.0.8802.1.1>(NA)
```

其它显示信息略……

表1-3 display snmp-agent mib-node 命令显示信息描述表

字段	描述
-std	MIB节点名
<1.0>	MIB节点对应的OID
(NA)	MIB节点访问权限，取值为： <ul style="list-style-type: none"><li>• NA：表示节点不可访问</li><li>• NF：表示节点支持告警</li><li>• RO：表示节点支持只读访问</li><li>• RW：表示节点支持读写访问</li><li>• RC：表示节点支持读写创建访问</li><li>• WO：表示节点支持只写访问</li></ul>
*	表示叶子节点或表节点

# 显示 SNMP 支持 MIB 节点细节信息。

```
<Sysname> display snmp-agent mib-node details
```

```
iso(1)(dot1xPaeSystemAuthControl)
  |-std(0)(dot1xPaeSystemAuthControl)
    |-iso8802(8802)(dot1xPaeSystemAuthControl)
      |-ieee802dot1(1)(dot1xPaeSystemAuthControl)
        |-ieee802dot1mibs(1)(dot1xPaeSystemAuthControl)
```

其它显示信息略……

表1-4 display snmp-agent mib-node details 命令显示信息描述表

字段	描述
-std	MIB节点名
(0)	MIB节点对应OID末位
(lldpMessageTxInterval)	MIB节点下一个叶子节点名
*	表示叶子节点或表节点

# 显示 SNMP 支持的 MIB 表名、索引节点名及对应的 OID。

<Sysname> display snmp-agent mib-node index-node

```
Table          |dot1xPaePortTable
Index          ||dot1xPaePortNumber
OID            ||| 1.0.8802.1.1.1.1.1.2.1.1
```

其它显示信息略……

表1-5 display snmp-agent mib-node index-node 命令显示信息描述表

字段	描述
Table	MIB表名
Index	MIB索引节点名
OID	MIB索引节点对应的OID

# 显示 SNMP 支持的 MIB 告警节点名及对应的 OID、告警绑定变量节点名及对应的 OID。

<Sysname> display snmp-agent mib-node trap-node

```
Name          |lldpRemTablesChange
OID           ||1.0.8802.1.1.2.0.0.1
Trap Object
Name          |||lldpStatsRemTablesInserts
OID           ||||1.0.8802.1.1.2.1.2.2
Name          |||lldpStatsRemTablesDeletes
OID           ||||1.0.8802.1.1.2.1.2.3
Name          |||lldpStatsRemTablesDrops
OID           ||||1.0.8802.1.1.2.1.2.4
Name          |||lldpStatsRemTablesAgeouts
OID           ||||1.0.8802.1.1.2.1.2.5
```

其它显示信息略……

表1-6 display snmp-agent mib-node trap-node 命令显示信息描述表

字段	描述
Name	MIB告警节点名
OID	MIB告警节点对应的OID
Trap Object	MIB告警绑定变量节点相关信息（其中Name表示告警绑定变量节点名，OID表示变量名节点对应的OID）

# 显示 SNMP 支持的 MIB 节点详细信息，包括节点名、OID、节点类型、访问权限、数据类型，对应 MOR 定义、父子兄弟节点信息等。

<Sysname> display snmp-agent mib-node verbose

```
Name          |iso
```

```

OID          || 1
Properties   || NodeType:   Other
              || AccessType: NA
              || DataType:   NA
              || MOR:        0x00000000
Parent       ||
First child  || std
Next leaf    || dot1xPaeSystemAuthControl
Next sibling  ||
其它显示信息略……

```

表1-7 display snmp-agent mib-node verbose 命令显示信息描述表

字段	描述
Name	MIB节点名
OID	MIB节点对应的OID
NodeType	MIB节点类型，取值为： <ul style="list-style-type: none"> <li>• Table: 表节点</li> <li>• Row: 表中行节点</li> <li>• Column: 表中列节点</li> <li>• Leaf: 叶子节点</li> <li>• Group: 组节点（叶子节点的父节点）</li> <li>• Trapnode: 告警节点</li> <li>• Other: 其他类型</li> </ul>
AccessType	MIB节点访问权限，取值为： <ul style="list-style-type: none"> <li>• NA: 表示节点不可访问</li> <li>• NF: 表示节点支持告警</li> <li>• RO: 表示节点支持只读访问</li> <li>• RW: 表示节点支持读写访问</li> <li>• RC: 表示节点支持读写创建访问</li> <li>• WO: 表示节点支持只写访问</li> </ul>

字段	描述
DataType	MIB节点数据类型，取值为： <ul style="list-style-type: none"> <li>Integer: 整数</li> <li>Integer32: 32 位整数</li> <li>Unsigned32: 32 位无符号整数</li> <li>Gauge: 可增可减的非负整数</li> <li>Gauge32: 32 位可增可减的非负整数</li> <li>Counter: 可增不可减的非负整数</li> <li>Counter32: 32 位可增不可减的非负整数</li> <li>Counter64: 64 位可增不可减的非负整数</li> <li>Timeticks: 用于计时的非负整数</li> <li>Octstring: 八进制字符串</li> <li>OID: 对象标识符</li> <li>IPAddress: 用于 IP 规范格式的 32 位地址</li> <li>Networkaddress: 网络 IP 地址</li> <li>Opaque: 任意数据</li> <li>Userdefined: 用户类型</li> <li>BITS: 所述位枚举</li> </ul>
MOR	MIB节点对应的MOR定义
Parent	父节点名
First child	第一个子节点名
Next leaf	下一个叶子节点名
Next sibling	右兄弟节点名
Allow	允许的操作类型，取值包括如下： <ul style="list-style-type: none"> <li>get/set/getnext: 允许所有操作</li> <li>get: 只允许 Get 操作</li> <li>set: 只允许 Set 操作</li> <li>getnext: 只允许 GetNext 操作</li> </ul>
Value range	节点的取值范围
Index	表索引，仅表节点显示此字段

### 1.1.6 display snmp-agent mib-view

**display snmp-agent mib-view** 命令用来显示 MIB 视图的信息。

#### 【命令】

**display snmp-agent mib-view** [ **exclude** | **include** | **viewname** *view-name* ]

## 【视图】

任意视图

## 【缺省用户角色】

network-admin  
network-operator

## 【参数】

**exclude:** 显示属性为 **exclude** 的 MIB 视图的信息。

**include:** 显示属性为 **include** 的 MIB 视图的信息。

**viewname** *view-name*: 显示指定名称 MIB 视图的信息，*view-name* 为视图的名称，为 1~32 个字符的字符串，区分大小写。

## 【使用指导】

不指定参数时，显示所有 MIB 视图的信息。

## 【举例】

# 显示设备的所有 MIB 视图。

```
<Sysname> display snmp-agent mib-view
```

```
View name: ViewDefault
  MIB Subtree: iso
  Subtree mask:
  Storage-type: nonVolatile
  View Type: included
  View status: active
```

```
View name: ViewDefault
  MIB Subtree: snmpUsmMIB
  Subtree mask:
  Storage-type: nonVolatile
  View Type: excluded
  View status: active
```

```
View name: ViewDefault
  MIB Subtree: snmpVacmMIB
  Subtree mask:
  Storage-type: nonVolatile
  View Type: excluded
  View status: active
```

```
View name: ViewDefault
  MIB Subtree: snmpModules.18
  Subtree mask:
  Storage-type: nonVolatile
  View Type: excluded
  View status: active
```

以上信息表明，设备上当前有四个 MIB 视图，名称均为 ViewDefault。使用 ViewDefault 视图名限制 NMS 访问时，除了 snmpUsmMIB、snmpVacmMIB、snmpModules.18 子树下的 MIB 对象，NMS 可以访问 iso 子树下其它所有 MIB 对象。

表1-8 display snmp-agent mib-view 命令显示信息描述表

字段	描述
View name	视图名
MIB Subtree	MIB视图对应的MIB子树
Subtree mask	MIB子树的掩码
Storage-type	存储方式，分为以下几种：volatile、nonVolatile、permanent、readOnly、other，具体请参见 <a href="#">表1-1</a>
View Type	MIB视图的类型（即该视图与MIB子树的关系），包括included和excluded两种： <ul style="list-style-type: none"> <li>included 表示当前视图包括该子树的所有节点，即可以访问子树内的所有 MIB 对象</li> <li>excluded 表示当前视图不包括该子树的任意节点，即子树内的所有 MIB 对象都不能被访问</li> </ul>
View status	MIB视图的状态，包括： <ul style="list-style-type: none"> <li>active 表示 MIB 视图可用</li> <li>inactive 表示 MIB 视图不可用。用户不能对处于该状态的 MIB 视图中的节点执行读写操作，但允许 MIB 视图中的节点发送 Trap 和 Inform 消息</li> </ul>

## 【相关命令】

- **snmp-agent mib-view**

## 1.1.7 display snmp-agent remote

**display snmp-agent remote** 命令用来显示远端 SNMP 实体的引擎 ID。

## 【命令】

**display snmp-agent remote** [ { *ipv4-address* | **ipv6** *ipv6-address* } ]

## 【视图】

任意视图

## 【缺省用户角色】

network-admin  
network-operator

## 【参数】

**ipv4-address**: 显示指定 IPv4 地址的远端 SNMP 实体的引擎 ID。*ipv4-address* 表示远端 SNMP 实体的 IPv4 地址。

**ipv6 ipv6-address**: 显示指定 IPv6 地址的远端 SNMP 实体的引擎 ID。*ipv6-address* 表示远端 SNMP 实体的 IPv6 地址。

【使用指导】

SNMP 实体引擎 ID 是 SNMP 实体的唯一标识，它在一个 SNMP 管理域内是唯一的。SNMP 实体引擎是 SNMP 实体的重要组成部分，完成 SNMP 信息的信息调度、信息处理、安全验证、访问控制等功能。

如果未指定远端 SNMP 实体的 IP 地址，则会显示设备上配置的所有远端 SNMP 实体的引擎 ID。

【举例】

# 显示设备上配置的所有远端 SNMP 实体的引擎 ID。

```
<Sysname> display snmp-agent remote
Remote engineID: 800063A28000A0FC00580400000001
IPv4 address: 1.1.1.1
```

表1-9 display snmp-agent remote 命令显示信息描述表

字段	描述
Remote engineID	远端SNMP实体的引擎，可通过 <b>snmp-agent remote</b> 命令配置
IPv4 address	远端SNMP实体的IPv4地址
IPv6 address	远端SNMP实体的IPv6地址。当配置 <b>snmp-agent remote</b> 命令时绑定的是IPv6地址时，显示该信息

【相关命令】

- **snmp-agent remote**

1.1.8 display snmp-agent statistics

**display snmp-agent statistics** 命令用来显示 SNMP 报文的统计信息。

【命令】

**display snmp-agent statistics**

【视图】

任意视图

【缺省用户角色】

network-admin  
network-operator

【举例】

# 显示 SNMP 报文的统计信息。

```
<Sysname> display snmp-agent statistics
1684 messages delivered to the SNMP entity.
5 messages were for an unsupported version.
0 messages used an unknown SNMP community name.
0 messages represented an illegal operation for the community supplied.
0 ASN.1 or BER errors in the process of decoding.
1679 messages passed from the SNMP entity.
```



```

0 SNMP PDUs had badValue error-status.
0 SNMP PDUs had genErr error-status.
0 SNMP PDUs had noSuchName error-status.
0 SNMP PDUs had tooBig error-status (Maximum packet size 1500).
16544 MIB objects retrieved successfully.
2 MIB objects altered successfully.
7 GetRequest-PDU accepted and processed.
7 GetNextRequest-PDU accepted and processed.
1653 GetBulkRequest-PDU accepted and processed.
1669 GetResponse-PDU accepted and processed.
2 SetRequest-PDU accepted and processed.
0 Trap PDUs accepted and processed.
0 alternate Response Class PDUs dropped silently.
0 forwarded Confirmed Class PDUs dropped silently.

```

表1-10 display snmp-agent statistics 命令显示信息描述表

字段	描述
messages delivered to the SNMP entity	Agent收到的数据报文个数
messages were for an unsupported version	版本不支持的数据报文个数
messages used an unknown SNMP community name	使用了非法团体名的数据报文个数
messages represented an illegal operation for the community supplied	包含了超出团体名权限的操作的数据报文个数
ASN.1 or BER errors in the process of decoding	在解码过程中发生ASN.1（Abstract Syntax Notation dot one，抽象记法1）或BER（Basic Encoding Rules，基本编码规则）错误的数据报文个数
messages passed from the SNMP entity	Agent发送给别的SNMP实体的数据报文个数
SNMP PDUs had badValue error-status	错误类型为BadValues的数据报文个数
SNMP PDUs had genErr error-status	genErr错误的数据报文个数
SNMP PDUs had noSuchName error-status	NoSuchName错误的数据报文个数
SNMP PDUs had tooBig error-status	TooBig错误的数据报文个数（设备允许通过的最大SNMP PDU为1500字节）
MIB objects retrieved successfully	已成功获取的MIB对象个数
MIB objects altered successfully	已成功修改的MIB对象个数
GetRequest-PDU accepted and processed	已接收并处理的Get请求的个数
GetNextRequest-PDU accepted and processed	已接收并处理的GetNext请求的个数
GetBulkRequest-PDU accepted and processed	已接收并处理的GetBulk请求的个数
GetResponse-PDU accepted and processed	已接收并处理的Get响应的个数
SetRequest-PDU accepted and processed	已接收并处理的Set请求的个数
Trap PDUs accepted and processed	已接收并处理的Trap和Inform报文的个数
alternate Response Class PDUs dropped silently	被丢弃的响应数据报文个数

字段	描述
forwarded Confirmed Class PDUs dropped silently	被丢弃的转发数据报文个数

### 1.1.9 display snmp-agent sys-info

**display snmp-agent sys-info** 命令用来显示 SNMP 设备的系统信息。

#### 【命令】

**display snmp-agent sys-info [ contact | location | version ] \***

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

**contact:** 显示当前设备维护者的联系信息。  
**location:** 显示当前设备的物理位置信息。  
**version:** 显示当前设备中运行的 SNMP 版本号。

#### 【使用指导】

不指定参数时，显示设备的全部系统信息。

#### 【举例】

```
# 显示设备系统信息。
<Sysname> display snmp-agent sys-info
The contact information of the agent:
    New H3C Technologies Co., Ltd.

The location information of the agent:
    Hangzhou, China

The SNMP version of the agent:
    SNMPv3
```

#### 【相关命令】

- **snmp-agent sys-info**

### 1.1.10 display snmp-agent trap queue

**display snmp-agent trap queue** 命令用来显示告警信息队列的基本信息。

#### 【命令】

**display snmp-agent trap queue**

【视图】

任意视图

【缺省用户角色】

network-admin  
network-operator

【举例】

# 显示当前告警信息队列的配置及使用情况。

```
<Sysname> display snmp-agent trap queue
Queue size: 100
Message number: 6
```

表1-11 display snmp-agent trap queue 命令显示信息描述表

字段	描述
Queue size	告警信息队列长度
Message number	告警信息队列中当前告警信息的个数

【相关命令】

- snmp-agent trap life
- snmp-agent trap queue-size

1.1.11 display snmp-agent trap-list

display snmp-agent trap-list 命令用来显示 SNMP 告警功能的开启状态。

【命令】

display snmp-agent trap-list

【视图】

任意视图

【缺省用户角色】

network-admin  
network-operator

【使用指导】

如果一个模块包含多个子模块，只要有任何一个子模块的告警信息是使能的，就显示整个模块是使能的。

业务模块是否支持 SNMP 告警功能请通过执行 **snmp-agent trap enable ?**命令来获知。本命令的显示信息和 **snmp-agent trap enable** 命令以及业务模块的配置相关。

【举例】

# 显示 SNMP 告警功能的开启状态。

```
<Sysname> display snmp-agent trap-list
arp notification is disabled.
```

```
configuration notification is enabled.
mac-address notification is enabled.
radius notification is disabled.
standard notification is enabled.
syslog notification is disabled.
system notification is enabled.

Enabled notifications: 4; Disabled notifications: 3
```

## 【相关命令】

- **snmp-agent trap enable**

### 1.1.12 display snmp-agent usm-user

**display snmp-agent usm-user** 命令用来显示 SNMPv3 用户信息。

## 【命令】

```
display snmp-agent usm-user [ engineid engineid | group group-name | username user-name ] *
```

## 【视图】

任意视图

## 【缺省用户角色】

```
network-admin
network-operator
```

## 【参数】

**engineid** *engineid*: 显示指定引擎 ID 的 SNMPv3 用户信息, *engineid* 表示 SNMP 引擎 ID, 不区分大小写。SNMPv3 用户创建的时候, 系统会记录当时设备的 SNMP 实体引擎 ID, 如果设备的引擎 ID 被修改, 则被创建的 SNMPv3 用户将暂时无效, 只有引擎 ID 恢复后, 才能继续生效。

**group** *group-name*: 显示属于指定 SNMP 组的 SNMPv3 用户信息, 区分大小写。

**username** *user-name*: 显示指定名称的 SNMPv3 用户信息, 区分大小写。

## 【使用指导】

使用 **snmp-agent usm-user** 命令可以创建 SNMPv1/v2c/v3 用户, 如果创建的是 SNMPv1/v2c 用户, 系统自动添加一个新的同名的团体名, 并将这个用户当成 SNMPv1/v2c 团体来处理。所以, 不能通过 **display snmp-agent usm-user** 命令来查看 SNMPv1/v2c 用户的信息, 能通过 **display snmp-agent community** 查看 SNMPv1/v2c 用户对应的团体的信息。

## 【举例】

# 显示设备上已创建的所有 SNMPv3 用户的信息。

```
<Sysname> display snmp-agent usm-user
Username: userv3
Group name: mygroupv3
Engine ID: 800063A203000FE240A1A6
Storage-type: nonVolatile
UserStatus: active
```

ACL: 2000

Username: userv3  
Group name: mygroupv3  
Engine ID: 8000259503000BB3100A508  
Storage-type: nonVolatile  
UserStatus: active  
ACL name: testacl

Username: userv3code  
Role name: groupv3code  
network-operator  
Engine ID: 800063A203000FE240A1A6  
Storage-type: nonVolatile  
UserStatus: active

Username: userv3code  
Role name: snmprole  
network-operator  
Engine ID: 800063A280000002BB0001  
Storage-type: nonVolatile  
UserStatus: active

表1-12 display snmp-agent usm-user 命令显示信息描述表

字段	描述
Username	SNMP用户的用户名
Group name	SNMP用户所在组的组名
Role name	SNMP用户的角色名称
Engine ID	SNMP用户创建时使用的SNMP实体引擎ID
Storage-type	存储方式，分为以下几种：volatile、nonVolatile、permanent、readOnly、other，具体请参见 <a href="#">表1-1</a>
UserStatus	SNMP用户的状态，分为以下几种： <ul style="list-style-type: none"><li>active: 有效</li><li>notInService: 当前不可用</li><li>notReady: 未配置完成</li><li>other: 其他</li></ul>
ACL	使用的ACL列表的编号（该字段仅在用户与ACL绑定后显示，不会与ACL name同时存在）
ACL name	使用的ACL列表的名称（该字段仅在用户与ACL名称绑定后显示，不会与ACL同时存在）

#### 【相关命令】

- snmp-agent usm-user v3

### 1.1.13 enable snmp trap updown

**enable snmp trap updown** 命令用来开启接口状态变化的告警功能。

**undo enable snmp trap updown** 命令用来关闭接口状态变化的告警功能。

#### 【命令】

```
enable snmp trap updown
undo enable snmp trap updown
```

#### 【缺省情况】

接口状态变化的告警功能处于开启状态。

#### 【视图】

接口视图

#### 【缺省用户角色】

network-admin

#### 【使用指导】

需要注意的是，如果要求接口在状态发生改变时生成接口状态变化的告警信息，需要开启全局告警功能并在接口开启接口状态变化的告警功能。接口下开启请使用命令 **enable snmp trap updown**，全局下开启请使用命令 **snmp-agent trap enable standard [ linkdown | linkup ] \***。

#### 【举例】

# 允许发送端口 GigabitEthernet1/0/1 的 linkUp/linkDown 的 SNMP 告警，使用团体名 public，向 IP 地址为 10.1.1.1 的目的主机发送 Trap 报文。

```
<Sysname> system-view
[Sysname] snmp-agent trap enable
[Sysname] snmp-agent target-host trap address udp-domain 10.1.1.1 params securityname public
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] enable snmp trap updown
```

#### 【相关命令】

- **snmp-agent target-host**
- **snmp-agent trap enable**

### 1.1.14 snmp-agent

**snmp-agent** 命令用来开启 SNMP Agent 功能。

**undo snmp-agent** 命令用来关闭 SNMP Agent 功能。

#### 【命令】

```
snmp-agent
undo snmp-agent
```

#### 【缺省情况】

SNMP Agent 功能处于关闭状态。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【使用指导】

执行除 **snmp-agent calculate-password** 外任何以 **snmp-agent** 开头的命令都可以开启 SNMP Agent 功能。

## 【举例】

# 开启设备的 SNMP Agent 功能。

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

```
[Sysname] snmp-agent
```

### 1.1.15 snmp-agent { inform | trap } source

**snmp-agent { inform | trap } source** 命令用来配置发送的告警信息的源 IP 地址。

**undo snmp-agent { inform | trap } source** 命令用来恢复缺省情况。

## 【命令】

**snmp-agent { inform | trap } source** *interface-type interface-number*

**undo snmp-agent { inform | trap } source**

## 【缺省情况】

使用出接口的 IP 地址作为告警信息的源 IP 地址。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**inform:** 用来指定 Inform 报文中的源 IP 地址。

**trap:** 用来指定 Trap 报文中的源 IP 地址。

*interface-type interface-number:* 指定三层接口类型与接口编号。

## 【使用指导】

执行该命令后，系统会使用指定接口的主 IP 地址作为发送出去的告警信息的源 IP 地址。这样，在 NMS 上就可以使用该 IP 地址唯一标志 Agent。即便 Agent 使用不同的出接口发送告警信息，NMS 都可以使用该 IP 地址来过滤 Agent 发送的所有告警信息。

在将某个接口配置为获取告警信息的源地址接口之前需要注意的是：

- 如果配置的接口已存在，并且配置了合法的 IP 地址，则该 IP 地址将作为告警信息的源地址；
- 如果配置的接口不存在，则该命令会配置失败；
- 如果配置的接口已存在，但未配置合法的 IP 地址，则该命令不生效，在接口配置了合法 IP 地址后，该命令会自动生效。

### 【举例】

# 配置 Trap 报文的源地址为以太网接口 GigabitEthernet1/0/1 上的接口主 IP 地址。

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

```
[Sysname] snmp-agent trap source gigabitethernet 1/0/1
```

# 配置 Inform 报文的源地址为以太网接口 GigabitEthernet1/0/2 上的接口主 IP 地址。

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

```
[Sysname] snmp-agent inform source gigabitethernet 1/0/2
```

### 【相关命令】

- **snmp-agent trap enable**
- **snmp-agent target-host**

## 1.1.16 snmp-agent calculate-password

**snmp-agent calculate-password** 命令用来为明文密码计算对应的密文密码。

### 【命令】

非 FIPS 模式下：

```
snmp-agent calculate-password plain-password mode { 3desmd5 | 3dessha |  
aes192md5 | aes192sha | aes256md5 | aes256sha | md5 | sha } { local-engineid |  
specified-engineid engineid }
```

FIPS 模式下：

```
snmp-agent calculate-password plain-password mode { aes192sha | aes256sha |  
sha } { local-engineid | specified-engineid engineid }
```

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**plain-password**：表示明文密码，为 1~64 个字符的字符串，区分大小写。

**mode**：指定认证算法和加密算法。设备支持的认证算法有 HMAC-MD5 和 HMAC-SHA1，其中 HMAC-MD5 的计算速度比 HMAC-SHA1 快，而 HMAC-SHA1 的安全强度比 HMAC-MD5 高；设备支持的加密算法安全性由高到低依次是：AES256、AES192、AES、3DES、DES，安全性高的加密算法实现机制复杂，运算速度慢。对于普通的安全要求，DES 算法就可以满足需要。

- **3desmd5**：用于计算密文加密密码，此时对应的认证算法必须为 HMAC-MD5，加密算法必须为 3DES。
- **3dessha**：用于计算密文加密密码，此时对应的认证算法必须为 HMAC-SHA1，加密算法必须为 3DES。
- **aes192md5**：用于计算密文加密密码，此时对应的认证算法必须为 HMAC-MD5，加密算法必须为 AES192。
- **aes192sha**：用于计算密文加密密码，此时对应的认证算法必须为 HMAC-SHA1，加密算法必须为 AES192。



- **aes256md5**: 用于计算密文加密密码, 此时对应的认证算法必须为 HMAC-MD5, 加密算法必须为 AES256。
- **aes256sha**: 用于计算密文加密密码, 此时对应的认证算法必须为 HMAC-SHA1, 加密算法必须为 AES256。
- **md5**: 用于计算密文认证密码和密文加密密码。用于计算密文认证密码时, 认证算法必须为 HMAC-MD5; 用于计算密文加密密码时, 认证算法必须为 HMAC-MD5, 加密算法可以为 AES 或者 DES。因为当认证算法为 HMAC-MD5 时, 相同的明文密码, 各自计算出来的密文认证密码和密文加密密码均相同。
- **sha**: 用于计算密文认证密码和密文加密密码。用于计算密文认证密码时, 认证算法必须为 HMAC-SHA1; 用于计算密文加密密码时, 认证算法必须为 HMAC-SHA1, 加密算法可以为 AES 或者 DES。因为当认证算法为 HMAC-SHA1 时, 相同的明文密码, 各自计算出来的密文认证密码和密文加密密码均相同。

**local-engineid**: 使用本地引擎 ID 计算密文密码, 引擎 ID 的相关描述与配置可参考命令 **snmp-agent local-engineid**。

**specified-engineid**: 使用用户指定的引擎 ID 计算密文密码。

**engineid**: 引擎 ID, 必须为偶数个十六进制数, 偶数的取值范围为 10~64, 不区分大小写。全 0 和全 F 均被认为是无效参数。

### 【使用指导】

执行本命令前, 必须先开启设备的 SNMP Agent 功能。

在创建 SNMPv3 用户时, 如果指明认证或者加密密码采用密文配置方式, 则可以输入该命令生成的密文密码来代替明文密码, 从而避免在非安全环境直接输入明文密码造成的安全隐患。

生成的密码是和引擎 ID 相关联的, 在某一引擎 ID 下生成的密码, 也只在此引擎 ID 下生效。

### 【举例】

# 使用本地引擎 ID 和 HMAC-SHA1 算法计算明文认证密码 authkey 的密文密码。

```
<Sysname> system-view
[Sysname] snmp-agent calculate-password authkey mode sha local-engineid
The encrypted key is: 09659EC5A9AE91BA189E5845E1DDE0CC
```

### 【相关命令】

- **snmp-agent local-engineid**
- **snmp-agent usm-user v3**

## 1.1.17 snmp-agent community

**snmp-agent community** 命令用来创建 SNMP 团体。

**undo snmp-agent community** 命令用来删除 SNMP 团体。

### 【命令】

VACM 方式:

```
snmp-agent community { read | write } [ simple | cipher ] community-name
[ mib-view view-name ] [ acl { ipv4-acl-number | name ipv4-acl-name } | acl ipv6
{ ipv6-acl-number | name ipv6-acl-name } ] *
```

```
undo snmp-agent community [ cipher ] community-name
```

RBAC 方式:

```
snmp-agent community [ simple | cipher ] community-name user-role role-name  
[ acl { ipv4-acl-number | name ipv4-acl-name } | acl ipv6 { ipv6-acl-number |  
name ipv6-acl-name } ] *
```

```
undo snmp-agent community [ cipher ] community-name
```

#### 【缺省情况】

不存在 SNMP 团体。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**read:** 表示对 MIB 对象的访问权限为只读。NMS 使用该团体名访问 Agent 时只能执行读操作。

**write:** 表示对 MIB 对象的访问权限为读写。NMS 使用该团体名访问 Agent 时可以执行读、写操作。

**simple:** 表示以明文方式配置团体名并以密文方式保存到配置文件中。

**cipher:** 表示以密文方式配置团体名并以密文方式保存到配置文件中。

**community-name:** 配置明文团体名或密文团体名，是限制 NMS 访问 Agent 时所使用的团体名。区分大小写，需要转义的字符请加 “\” 后输入。当以明文方式配置时，团体名为 1~32 个字符的字符串；当以密文方式配置时，团体名为 33~73 个字符的字符串。

**mib-view view-name:** 用来指定 NMS 可以访问的 MIB 对象的范围，**view-name** 表示 MIB 视图名，为 1~32 个字符的字符串，区分大小写。不指定参数时，缺省视图为 ViewDefault。

**user-role role-name:** 该团体对应的角色名称，**role-name** 为 1~63 个字符的字符串，区分大小写。

**acl:** 将团体名与基本/高级 IPv4 ACL 绑定。

**ipv4-acl-number:** 表示基本或高级 IPv4 ACL 的编号，其中基本 IPv4 ACL 的取值范围为 2000~2999，高级 IPv4 ACL 的取值范围为 3000~3999。

**name ipv4-acl-name:** 表示基本或高级 IPv4 ACL 的名称，为 1~63 个字符的字符串，不区分大小写。

**acl ipv6:** 将团体名与基本/高级 IPv6 ACL 绑定。

**ipv6-acl-number:** 表示基本或高级 IPv6 ACL 的编号，其中基本 IPv6 ACL 的取值范围为 2000~2999，高级 IPv6 ACL 的取值范围为 3000~3999。

**name ipv6-acl-name:** 表示基本或高级 IPv6 ACL 的名称，为 1~63 个字符的字符串，不区分大小写。

#### 【使用指导】

FIPS 模式下，不支持本命令。

为了安全起见,只有具有 **network-admin** 或者 **level-15** 用户角色的用户登录设备后才能执行本命令。其它角色的用户,即使授权了 **SNMP** 特性或本命令的操作权限,也不能执行本命令。

该命令用于 **SNMPv1** 和 **SNMPv2c** 组网环境。

团体是 **NMS** 和 **Agent** 的集合,用团体名来标志。团体名相当于密码,团体内的设备通信使用团体名来进行认证。只有 **NMS** 和 **Agent** 上配置的团体名相同时,才能互相访问。通常情况下,“**public**”被用来作为读权限团体名、“**private**”被用来作为写权限团体名。为了增强安全性,网络管理员也可以配置其它团体名。

用户有两种方式创建团体:

- 使用 **snmp-agent community** 命令来创建团体。
- 配置 **snmp-agent usm-user { v1 | v2c }** 和 **snmp-agent group { v1 | v2c }** 命令成功创建 **SNMPv1** 或 **SNMPv2c** 用户以及相应的组后,系统会以用户名为团体名自动创建一个团体。

**display snmp-agent community** 会显示这两种方式创建的团体的信息。

使用 **snmp-agent community** 命令创建团体时,可以通过两种配置方式来控制团体的访问:

- **VACM** (**View-based Access Control Model**, 基于视图的访问控制模型)的配置方式:该方式通过指定 **MIB** 视图来限定 **NMS** 可以访问的 **MIB** 节点,访问的动作包括只读和读写两种。
- **RBAC** (**Role Based Access Control**, 基于角色的访问控制)的配置方式:该方式使用用户角色来限定 **NMS** 的访问权限。缺省情况下,用户角色 **network-admin** 和 **level-15** 可以读写所有的 **MIB** 节点, **network-operator** 可以只读所有的 **MIB** 节点。有关用户角色的详细信息,请参见“基础配置指导”中的“**RBAC**”。

推荐使用 **RBAC** 配置方式,安全性更高。

多次执行本命令最多可创建 10 个团体名。

多次执行该命令,指定的团体名相同,其它参数不同时,新配置生效。

不指定 **simple** 和 **cipher** 参数时,表示以明文方式配置团体名,并以明文方式保存到配置文件。

使用 **acl** 参数可以限制非法 **NMS** 访问设备:

- 当未引用 **ACL**、引用的 **ACL** 不存在、或者引用的 **ACL** 下没有配置规则时,允许所有 **NMS** 访问设备。
- 当引用的 **ACL** 下配置了规则时,则只有规则中 **permit** 的 **NMS** 才能访问设备,其他 **NMS** 不允许访问设备。

关于 **ACL** 的详细描述和介绍请参见“**ACL** 和 **QoS** 配置指导”中的“**ACL**”。

## 【举例】

# 以明文方式创建 **SNMP** 团体 **readaccess**, 并且允许 **NMS** 使用该团体名对设备上缺省视图内的 **MIB** 对象进行只读访问。

```
<Sysname> system-view
[Sysname] snmp-agent sys-info version v1 v2c
[Sysname] snmp-agent community read simple readaccess
```

# 以明文方式配置团体名 **writeaccess**, 并且只允许 IP 地址为 1.1.1.1 的 **NMS** 使用该团体名对设备上缺省视图内的 **MIB** 对象进行读写操作, 禁止其它 **NMS** 使用该团体名执行写操作。

```
<Sysname> system-view
[Sysname] acl basic 2001
[Sysname-acl-ipv4-basic-2001] rule permit source 1.1.1.1 0.0.0.0
```

```

[Sysname-acl-ipv4-basic-2001] rule deny source any
[Sysname-acl-ipv4-basic-2001] quit
[Sysname] snmp-agent sys-info version v2c
[Sysname] snmp-agent community write simple writeaccess acl 2001
# 以明文方式配置团体名 writeaccess，并且只允许 IP 地址为 1.1.1.2 的 NMS 使用该团体名对设备上缺省视图内的 MIB 对象进行读写操作，禁止其它 NMS 使用该团体名执行写操作。
<Sysname> system-view
[Sysname] acl basic name testacl
[Sysname-acl-ipv4-basic-testacl] rule permit source 1.1.1.2 0.0.0.0
[Sysname-acl-ipv4-basic-testacl] rule deny source any
[Sysname-acl-ipv4-basic-testacl] quit
[Sysname] snmp-agent sys-info version v2c
[Sysname] snmp-agent community write simple writeaccess acl name testacl
# 以明文方式创建团体名 wr-sys-acc，使用该团体名访问设备时只能对 system(OID 为 1.3.6.1.2.1.1) 子树下的 MIB 对象执行写操作。
<Sysname> system-view
[Sysname] snmp-agent sys-info version v1 v2c
[Sysname] undo snmp-agent mib-view ViewDefault
[Sysname] snmp-agent mib-view included test system
[Sysname] snmp-agent community write simple wr-sys-acc mib-view test

```

#### 【相关命令】

- **display snmp-agent community**
- **snmp-agent mib-view**

### 1.1.18 snmp-agent community-map

**snmp-agent community-map** 命令用来创建团体名到 SNMP 上下文的映射。

**undo snmp-agent community-map** 命令用来删除团体名到 SNMP 上下文的映射。

#### 【命令】

```

snmp-agent community-map community-name context context-name
undo snmp-agent community-map community-name context context-name

```

#### 【缺省情况】

不存在团体名到 SNMP 上下文的映射。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*community-name*: 团体名，为 1~32 个字符的字符串，区分大小写。

*context-name*: SNMP 上下文。为 1~32 个字符的字符串，区分大小写。

### 【使用指导】

用户配置成功后，使用 SNMP v1/v2 版本连接 SNMP Agent 时，SNMP 插件端所获取的上下文，是此时 NMS 访问 Agent，使用的团体名映射的上下文。如团体名未配置上下文映射，则获取不到。系统中可配置的映射最多为 10 个。

### 【举例】

# 配置一个团体名到 SNMP 上下文的映射。

```
<Sysname> system-view
[Sysname] snmp-agent community-map private context testcontext
```

### 【相关命令】

- **display snmp-agent community**

## 1.1.19 snmp-agent context

**snmp-agent context** 命令用来创建 SNMP 上下文。

**undo snmp-agent context** 命令用来删除 SNMP 上下文。

### 【命令】

```
snmp-agent context context-name
undo snmp-agent context context-name
```

### 【缺省情况】

不存在 SNMP 上下文。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**context-name**: SNMP 上下文，为 1~32 个字符的字符串，区分大小写。

### 【使用指导】

NMS 未配置上下文，或 NMS 与 Agent 配置为相同的上下文时，两者可以连接成功，否则返回超时。系统中可配置的 SNMP 上下文最多为 20 个。

### 【举例】

# 创建一个新的 context。

```
<Sysname> system-view
[Sysname] snmp-agent context testcontext
```

### 【相关命令】

- **display snmp-agent context**

## 1.1.20 snmp-agent group

**snmp-agent group** 命令用来创建 SNMP 组。

**undo snmp-agent group** 命令用来删除 SNMP 组。

## 【命令】

非 FIPS 模式下：

- SNMPv1 和 SNMPv2c 版本下的命令格式是：

```
snmp-agent group { v1 | v2c } group-name [ read-view view-name ]  
[ write-view view-name ] [ notify-view view-name ] [ acl { ipv4-acl-number  
| name ipv4-acl-name } | acl ipv6 { ipv6-acl-number | name  
ipv6-acl-name } ] *  
undo snmp-agent group { v1 | v2c } group-name
```

- SNMPv3 版本下的命令格式是：

```
snmp-agent group v3 group-name [ authentication | privacy ] [ read-view  
read-view ] [ write-view write-view ] [ notify-view notify-view ] [ acl  
{ ipv4-acl-number | name ipv4-acl-name } | acl ipv6 { ipv6-acl-number |  
name ipv6-acl-name } ] *  
undo snmp-agent group v3 group-name [ authentication | privacy ]
```

FIPS 模式下：

```
snmp-agent group v3 group-name { authentication | privacy } [ read-view  
read-view ] [ write-view write-view ] [ notify-view notify-view ] [ acl  
{ ipv4-acl-number | name ipv4-acl-name } | acl ipv6 { ipv6-acl-number | name  
ipv6-acl-name } ] *  
undo snmp-agent group v3 group-name { authentication | privacy }
```

## 【缺省情况】

不存在 SNMP 组。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**v1**：SNMPv1 版本。

**v2c**：SNMPv2c 版本。

**v3**：SNMPv3 版本。

**group-name**：SNMP 组的名称，取值范围为 1～32 个字符的字符串，区分大小写。

**authentication**：表示对报文进行认证但不加密。

**privacy**：表示对报文进行认证和加密。

**read-view view-name**：只读视图名，为 1～32 个字符的字符串，区分大小写。缺省值为 ViewDefault。

**write-view view-name**：读写视图名，为 1～32 个字符的字符串，区分大小写。缺省情况下，未配置读写视图，即 NMS 不能对设备的所有 MIB 对象进行写操作。

**notify-view view-name**: 可以发告警信息的视图名, 为 1~32 个字符的字符串, 区分大小写。缺省情况下, 未配置告警信息视图。

**acl**: 将组与基本/高级 IPv4 ACL 绑定。

**ipv4-acl-number**: 表示基本或高级 IPv4 ACL 的编号, 其中基本 IPv4 ACL 编号的取值范围为 2000~2999, 高级 IPv4 ACL 编号的取值范围为 3000~3999。

**name ipv4-acl-name**: 表示基本或高级 IPv4 ACL 的名称, 为 1~63 个字符的字符串, 不区分大小写。

**acl ipv6**: 将组与基本/高级 IPv6 ACL 绑定。

**ipv6-acl-number**: 表示基本或高级 IPv6 ACL 的编号, 其中基本 IPv4 ACL 编号的取值范围为 2000~2999, 高级 IPv6 ACL 编号的取值范围为 3000~3999。

**name ipv6-acl-name**: 表示基本或高级 IPv6 ACL 的名称, 为 1~63 个字符的字符串, 不区分大小写。

### 【使用指导】

FIPS 模式下, 不支持 SNMPv1 和 SNMPv2c 版本下的本命令。

为了安全起见, 只有具有 **network-admin** 或者 **level-15** 用户角色的用户登录设备后才能执行本命令。其它角色的用户, 即使授权了 **SNMP** 特性或本命令的操作权限, 也不能执行本命令。

**SNMP** 组可以定义安全模式、视图权限等信息, 配置在此组内的用户都具有这些公共属性。

系统中可配置的 **SNMP** 组最多为 20 个。

是否指定 **authentication** 和 **privacy** 参数存在如下情况:

- 当不指定 **authentication** 和 **privacy** 时, 表示不认证不加密。此时, 使用与该组绑定的用户名建立 **SNMP** 连接时, 均不认证不加密。即使用户配置了认证密码/加密密码, 认证密码/加密密码也不生效。
- 当指定 **authentication** 时, 表示认证不加密。此时, 使用与该组绑定的用户名建立 **SNMP** 连接时, 均认证不加密。即使用户配置了加密密码, 加密密码也不生效。
- 当指定 **privacy** 时, 表示认证加密。此时, 使用与该组绑定的用户名建立 **SNMP** 连接时, 均认证加密。该组内的用户必须配置认证密码和加密密码, 否则, 不能建立 **SNMP** 连接。

创建 **SNMP** 组和用户的时候都可以使用 **acl** 参数限制非法 **NMS** 访问设备, 只有两个 **ACL** 均允许的 **NMS** 才能访问设备。在创建组或用户时, **ACL** 均遵循以下规则:

- 当未引用 **ACL**、引用的 **ACL** 不存在、或者引用的 **ACL** 下没有配置规则时, 允许所有 **NMS** 访问设备。
- 当引用的 **ACL** 下配置了规则时, 则只有规则中 **permit** 的 **NMS** 才能访问设备, 其他 **NMS** 不允许访问设备。

关于 **ACL** 的详细描述和介绍请参见“**ACL** 和 **QoS** 配置指导”中的“**ACL**”。

### 【举例】

# 在运行 **SNMPv3** 版本的设备上创建一个 **SNMP** 组 **group1**。

```
<Sysname> system-view
[Sysname] snmp-agent group v3 group1
```

### 【相关命令】

- **display snmp-agent group**



- `snmp-agent mib-view`
- `snmp-agent usm-user`

### 1.1.21 snmp-agent local-engineid

`snmp-agent local-engineid` 命令用来配置本设备的 SNMP 引擎 ID。

`undo snmp-agent local-engineid` 命令用来恢复缺省情况。

#### 【命令】

```
snmp-agent local-engineid engineid
undo snmp-agent local-engineid
```

#### 【缺省情况】

设备引擎 ID 为公司的“企业号+设备信息”。每台设备的设备信息不同，请以设备实际情况为准。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*engineid*: SNMP 引擎 ID，必须为偶数个十六进制数，偶数的取值范围为 10~64，不区分大小写。全 0 和全 F 均被认为是无效参数。

#### 【使用指导】

SNMP 引擎 ID 有两个作用：

- 在 NMS 管理的所有设备中，每一台设备都需要用一个唯一的引擎 ID 来标识 Agent，缺省情况下每个设备有一个缺省的引擎 ID，网络管理员需要确保管理域内不能有重复的引擎 ID。
- SNMPv3 版本的用户名、密文密码等都和引擎 ID 相关联，如果更改了引擎 ID，则原引擎 ID 下配置的用户名、密码失效。

通常情况下，使用设备的缺省引擎 ID 即可，用户也可以根据网络整体规划给设备配置方便记忆的引擎 ID，比如 A 栋一楼的一号设备可以将它的引擎 ID 设置为 000Af0010001，二号设备可以配置为 000Af0010002。

#### 【举例】

```
# 配置本设备的 SNMP 引擎 ID 为 123456789A。
<Sysname> system-view
[Sysname] snmp-agent local-engineid 123456789A
```

#### 【相关命令】

- `display snmp-agent local-engineid`
- `snmp-agent usm-user`

### 1.1.22 snmp-agent log

`snmp-agent log` 命令用来开启 SNMP 日志功能。



**undo snmp-agent log** 命令用来关闭 SNMP 日志功能。

#### 【命令】

```
snmp-agent log { all | authfail | get-operation | set-operation }  
undo snmp-agent log { all | authfail | get-operation | set-operation }
```

#### 【缺省情况】

SNMP 日志功能处于关闭状态。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**all**: 表示 SNMP Get 操作、Set 操作和 SNMP 认证失败的日志开关。

**authfail**: 表示 SNMP 认证失败的日志开关。

**get-operation**: 表示 SNMP Get 操作的日志开关。

**set-operation**: 表示 SNMP Set 操作的日志开关。

#### 【使用指导】

当打开 SNMP Get 或者 Set 操作的日志开关，NMS 对 Agent 执行指定的操作时，Agent 会记录与该操作相关信息并保存到设备的信息中心。当打开 SNMP 认证失败的日志开关时，如果 Agent 收到来自 NMS 的 SNMP 请求但是没有通过认证，Agent 会记录相关信息并保存到设备的信息中心。保存的相关信息将通过信息中心的参数配置，最终决定 SNMP 日志的输出规则（即是否允许输出以及输出方向）。

#### 【举例】

# 打开 SNMP Get 操作的日志开关。

```
<Sysname> system-view  
[Sysname] snmp-agent log get-operation
```

# 打开 SNMP Set 操作的日志开关。

```
<Sysname> system-view  
[Sysname] snmp-agent log set-operation
```

# 打开 SNMP 认证失败的日志开关。

```
<Sysname> system-view  
[Sysname] snmp-agent log authfail
```

### 1.1.23 snmp-agent mib-view

**snmp-agent mib-view** 命令用来创建或者更新 MIB 视图。

**undo snmp-agent mib-view** 命令用来删除 MIB 视图。

#### 【命令】

```
snmp-agent mib-view { excluded | included } view-name oid-tree [ mask  
mask-value ]
```

**undo snmp-agent mib-view** *view-name*

#### 【缺省情况】

存在四个 MIB 视图，名称均为 ViewDefault:

- 视图一包含 MIB 子树 iso;
- 视图二不包含子树 snmpUsmMIB;
- 视图三不包含子树 snmpVacmMIB;
- 视图四不包含子树 snmpModules.18。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**excluded:** 表示当前视图不包括该 MIB 子树的任何节点（即禁止访问 MIB 子树的所有节点）。

**included:** 表示当前视图包括该 MIB 子树的所有节点（即允许访问 MIB 子树的所有节点）。

**view-name:** 视图名，为 1~32 个字符的字符串，区分大小写。

**oid-tree:** MIB 子树，用子树根节点的 OID（如“1.3.6.1.2.1.1”）或名称（如“system”）表示，为 1~255 个字符的字符串，区分大小写。OID 是由一系列的整数组成，标明节点在 MIB 树中的位置，它能唯一地标识一个 MIB 库中的对象。

**mask mask-value:** 对象子树的掩码，必须为偶数个十六进制数，偶数的取值范围为 1~32，不区分大小写。

#### 【使用指导】

MIB 视图是 MIB 的子集，由视图名和 MIB 子树来唯一确定一个 MIB 视图。视图名相同但包含的子树不同，则认为是不同的视图。

缺省视图可以通过 **display snmp-agent mib-view** 命令来查看。如果使用缺省视图限制 NMS 的访问权限时，除了 snmpUsmMIB、snmpVacmMIB、snmpModules.18 子树下的 MIB 对象，NMS 可以访问 iso 子树下其它所有 MIB 对象。缺省视图可以通过 **undo snmp-agent mib-view** 命令删除，但是删除以后，可能导致不能对 Agent 的所有 MIB 节点执行读写操作，除非另外手工配置视图。

#### 【举例】

# 创建并更新 MIB 视图信息，视图名称为 mibtest，先创建一个包含 mib-2 子树(OID 为“1.3.6.1.2.1”)所有对象的 MIB 视图，再更新为不包含“system”子树(OID 为“1.3.6.1.2.1.1”)所有对象的 MIB 视图。

```
<Sysname> system-view
[Sysname] snmp-agent sys-info version v1
[Sysname] snmp-agent mib-view included mibtest 1.3.6.1.2.1
[Sysname] snmp-agent mib-view excluded mibtest system
[Sysname] snmp-agent community read public mib-view mibtest
```

以上配置成功后，当 NMS 使用 SNMPv1 版本，public 团体名访问设备时，不能查询 system 子树的所有对象（比如 sysDescr 和 sysObjectID 等节点），可以查询 mib-2 子树下的其它所有对象。

### 【相关命令】

- `display snmp-agent mib-view`
- `snmp-agent group`

#### 1.1.24 snmp-agent packet max-size

`snmp-agent packet max-size` 命令用来配置 Agent 能接收或发送的 SNMP 报文的最大长度。

`undo snmp-agent packet max-size` 命令用来恢复缺省情况。

### 【命令】

`snmp-agent packet max-size byte-count`

`undo snmp-agent packet max-size`

### 【缺省情况】

Agent 能处理的 SNMP 报文的最大长度为 1500。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*byte-count*: Agent 能接收/发送的 SNMP 报文的最大长度，取值范围为 484~17940，单位为字节。

### 【使用指导】

配置报文的最大长度是为了防止网络中存在不支持分片的主机，而导致超长数据被丢弃。通常情况下，使用缺省值即可。

### 【举例】

# 配置 Agent 能接收/发送的 SNMP 报文的最大长度为 1024 字节。

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

```
[Sysname] snmp-agent packet max-size 1024
```

#### 1.1.25 snmp-agent packet response dscp

`snmp-agent packet response dscp` 命令用来配置 SNMP 响应报文的 DSCP（Differentiated Services Code Point，差分服务代码点）优先级。

`undo snmp-agent packet response dscp` 命令用来恢复缺省情况。

### 【命令】

`snmp-agent packet response dscp dscp-value`

`undo snmp-agent packet response dscp`

### 【缺省情况】

SNMP 响应报文的 DSCP 优先级为 0。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*dscp-value*: 表示 SNMP 响应报文的 DSCP 优先级，取值范围为 0~63。值越大表示优先级越高。

### 【使用指导】

DSCP 优先级封装在 IP 报文中的 ToS 字段，用来表示报文自身的优先级，设备可根据该优先级决定报文传输的优先程度。

### 【举例】

# 设置 SNMP 响应报文的 DSCP 优先级为 40。

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

```
[Sysname] snmp-agent packet response dscp 40
```

## 1.1.26 snmp-agent port

**snmp-agent port** 命令用来配置设备上接收 SNMP 报文的端口号。

**undo snmp-agent port** 命令用来恢复缺省情况。

### 【命令】

```
snmp-agent port port-number
```

```
undo snmp-agent port
```

### 【缺省情况】

使用 161 号端口接收 SNMP 报文。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*port-number*: 设备上接收 SNMP 报文的本地端口号，取值范围为 1~65535，缺省值为 161。

### 【使用指导】

用户配置成功后，使用新端口重新连接设备后，可以进行 Get/Set 等操作，此时使用 **display current-configuration** 命令查看 SNMP 相关配置，此项配置可以显示。

### 【举例】

# 指定新的端口号。

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

```
[Sysname] snmp-agent port 5555
```

# 恢复默认端口号。

```
<Sysname> system-view
[Sysname] undo snmp-agent port
```

### 1.1.27 snmp-agent remote

**snmp-agent remote** 命令用来配置远端 SNMP 实体的引擎 ID。

**undo snmp-agent remote** 命令用来取消已配置的远端 SNMP 实体的引擎 ID。

#### 【命令】

```
snmp-agent remote { ipv4-address | ipv6 ipv6-address } engineid engineid
undo snmp-agent remote ip-address
```

#### 【缺省情况】

未配置远端 SNMP 实体的引擎 ID。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**ipv4-address**: 远端 SNMP 实体的 IP 地址。

**ipv6 ipv6-address**: 远端 SNMP 实体的 IPv6 地址。

**engineid**: 引擎 ID, 必须为偶数个十六进制数, 偶数的取值范围为 10~64, 不区分大小写。全 0 和全 F 均被认为是无效参数。

#### 【使用指导】

当设备需要向 NMS 发送 SNMPv3 Inform 报文时, 必须配置该命令, 并将 *ip-address* 配置为 NMS 的 IP 地址, *engineid* 配置为 NMS 的引擎 ID。因为协议要求 SNMPv3 Inform 报文中必须携带一个权威引擎 ID, NMS 收到该报文后, 会用自己的引擎 ID 和这个权威引擎 ID 比较, 如果相同, 才能接收。

用户最多可以配置 20 个远端 SNMP 实体引擎 ID。

#### 【举例】

# 配置 IP 地址为 10.1.1.1 的 SNMP 实体的引擎 ID 为 123456789A。

```
<Sysname> system-view
[Sysname] snmp-agent remote 10.1.1.1 engineid 123456789A
```

#### 【相关命令】

- **display snmp-agent remote**

### 1.1.28 snmp-agent sys-info contact

**snmp-agent sys-info contact** 命令用来配置设备的维护联系信息。

**undo snmp-agent sys-info contact** 命令用来恢复缺省情况。

### 【命令】

```
snmp-agent sys-info contact sys-contact  
undo snmp-agent sys-info contact
```

### 【缺省情况】

设备的维护联系信息为 New H3C Technologies Co., Ltd.

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*sys-contact*: 描述系统维护联系信息，为 1~255 个字符的字符串，区分大小写。

### 【使用指导】

如果设备发生故障，设备维护人员可以利用系统维护联系信息，及时与设备生产厂商取得联系。

### 【举例】

```
# 配置设备的维护联系信息为 Dial System Operator # 27345。  
<Sysname> system-view  
[Sysname] snmp-agent sys-info contact Dial System Operator # 27345
```

### 【相关命令】

- **display snmp-agent sys-info**

## 1.1.29 snmp-agent sys-info location

**snmp-agent sys-info location** 命令用来配置设备的物理位置信息。

**undo snmp-agent sys-info location** 命令用来恢复缺省情况。

### 【命令】

```
snmp-agent sys-info location sys-location  
undo snmp-agent sys-info location
```

### 【缺省情况】

物理位置信息为 Hangzhou, China。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*sys-location*: 设备的物理位置信息，为 1~255 个字符的字符串，区分大小写。

## 【使用指导】

为便于识别和管理设备，请使用该命令将设备所处的物理位置记录在设备中。

## 【举例】

# 配置设备的物理位置信息为 Room524-row1-3。

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

```
[Sysname] snmp-agent sys-info location Room524-row1-3
```

## 【相关命令】

- **display snmp-agent sys-info**

### 1.1.30 snmp-agent sys-info version

**snmp-agent sys-info version** 命令用来配置设备支持的 SNMP 版本。

**undo snmp-agent sys-info version** 命令用来取消对指定 SNMP 版本的支持。

## 【命令】

非 FIPS 模式下：

```
snmp-agent sys-info version { all | { v1 | v2c | v3 } * }
```

```
undo snmp-agent sys-info version { all | { v1 | v2c | v3 } * }
```

FIPS 模式下：

```
snmp-agent sys-info version v3
```

```
undo snmp-agent sys-info version v3
```

## 【缺省情况】

启用 SNMPv3 版本。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**all**：支持 SNMPv1、SNMPv2c 和 SNMPv3 版本。

**v1**：支持 SNMPv1 版本。

**v2c**：支持 SNMPv2c 版本。

**v3**：支持 SNMPv3 版本。

## 【使用指导】

FIPS 模式下，不支持配置 SNMPv1 和 SNMPv2c 版本。

支持指定的 SNMP 版本后，设备才能收发该版本的 SNMP 报文。只有 NMS 和 Agent 使用的 SNMP 版本相同，NMS 才能和 Agent 建立连接。

## 【举例】

# 配置设备支持 SNMPv3 版本。

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

[Sysname] snmp-agent sys-info version v3

#### 【相关命令】

- **display snmp-agent sys-info**

### 1.1.31 snmp-agent target-host

**snmp-agent target-host** 命令用来配置接收 SNMP 告警信息的目的主机（能够解析 Trap 和 Inform 报文的设备，通常为 NMS）的属性。

**undo snmp-agent target-host** 命令用来取消目的主机配置。

#### 【命令】

非 FIPS 模式下：

```
snmp-agent target-host inform address udp-domain { ipv4-address | ipv6  
ipv6-address } [ udp-port port-number ] params securityname security-string  
{ v2c | v3 [ authentication | privacy ] }
```

```
snmp-agent target-host trap address udp-domain { ipv4-address | ipv6  
ipv6-address } [ udp-port port-number ] [ dscp dscp-value ] params  
securityname security-string [ v1 | v2c | v3 [ authentication | privacy ] ]
```

```
undo snmp-agent target-host { trap | inform } address udp-domain  
{ ipv4-address | ipv6 ipv6-address } params securityname security-string
```

FIPS 模式下：

```
snmp-agent target-host inform address udp-domain { ipv4-address | ipv6  
ipv6-address } [ udp-port port-number ] params securityname security-string  
v3 { authentication | privacy }
```

```
snmp-agent target-host trap address udp-domain { ipv4-address | ipv6  
ipv6-address } [ udp-port port-number ] [ dscp dscp-value ] params  
securityname security-string v3 { authentication | privacy }
```

```
undo snmp-agent target-host { trap | inform } address udp-domain  
{ ipv4-address | ipv6 ipv6-address } params securityname security-string
```

#### 【缺省情况】

未配置告警主机。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**inform**：配置接收 Inform 报文的参数的目的主机的参数。

**trap**：配置接收 Trap 报文的参数的目的主机的参数。

**address**：指定接收告警信息的目的主机的地址。

**udp-domain**：指定使用 UDP 协议来传输 SNMP 告警信息。



**ipv4-address**: 接收告警信息的目的主机的 IPv4 地址或主机名, 主机名为 1~253 个字符的字符串, 不区分大小写, 字符串仅可包含字母、数字、“-”、“\_”或“.”。若使用主机名配置, 发送时将获取主机名对应的 IPv4 地址, 向对应的主机发送告警信息。

**ipv6 ipv6-address**: 接收告警信息的目的主机的 IPv6 地址或主机名, 主机名为 1~253 个字符的字符串, 不区分大小写, 字符串仅可包含字母、数字、“-”、“\_”或“.”。若使用主机名配置, 发送时将获取主机名对应的 IPv6 地址, 向对应的主机发送告警信息。若使用 IPv6 地址配置, 则不能为链路本地地址。

**udp-port port-number**: 指定目的主机上用来接收告警信息的端口号, 缺省值为 162。

**dscp dscp-value**: 配置向目的主机发送的 Trap 报文的 DSCP 优先级, 取值范围为 0~63, 缺省值为 0。值越大表示优先级越高。DSCP 优先级封装在 IP 报文中的 ToS 字段, 用来表示报文自身的优先级, 设备可根据该优先级决定报文传输的优先程度。

**params securityname security-string**: 指定认证的参数, *security-string* 为 SNMPv1、SNMPv2c 的团体名或 SNMPv3 的用户名, 为 1~32 个字符的字符串, 区分大小写。

**v1**: SNMPv1 版本。

**v2c**: SNMPv2c 版本。

**v3**: SNMPv3 版本。

**authentication**: 指明对报文进行认证但不加密。认证功能用来验证报文的完整性或报文是否被篡改等, 认证密码在创建 SNMPv3 用户时配置。

**privacy**: 指明对报文进行认证和加密。加密是对报文的数据部分进行加密处理以防信息被窃取, 认证密码和加密密码在创建 SNMPv3 用户时配置。

## 【使用指导】

根据实际组网需要, 用户可以多次使用该命令配置不同的目的主机的属性, 使得设备可以向多个 NMS 发送告警信息。

不指定 **udp-port port-number** 参数时, 使用的端口号为 162。162 是 SNMP 协议规定的 NMS 接收告警信息的端口, 通常情况下 (比如使用 iMC 或者 MIB Browser 作为 NMS 时), 使用该缺省值即可。如果要将该参数修改为其它值, 则必须和 NMS 上的配置保持一致。

不指定 **v1**、**v2c**、**v3** 版本参数时, 使用的版本是 v1。设备配置的 SNMP 版本必须和 NMS 上运行的 SNMP 版本一致, 否则, NMS 将收不到告警信息。

不指定 **authentication** 和 **privacy** 参数时, 使用的是不认证不加密的安全级别。

## 【举例】

# 允许向 10.1.1.1 发送 SNMPv3 Trap 报文, 用户名为 public。

```
<Sysname> system-view
[Sysname] snmp-agent trap enable standard
[Sysname] snmp-agent target-host trap address udp-domain 10.1.1.1 params securityname public v3
```

## 【相关命令】

- **snmp-agent { inform | trap } source**
- **snmp-agent trap enable**
- **snmp-agent trap life**

### 1.1.32 snmp-agent trap enable

**snmp-agent trap enable** 命令用来开启 SNMP 告警功能。

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

#### 【命令】

```
snmp-agent trap enable [ configuration | protocol | standard [ authentication  
| coldstart | linkdown | linkup | warmstart ] * | system ]  
undo snmp-agent trap enable [ configuration | protocol | standard  
[ authentication | coldstart | linkdown | linkup | warmstart ] * | system ]
```

#### 【缺省情况】

SNMP 配置告警、标准告警和系统告警功能处于开启状态，其他各模块告警功能是否开启请参见各模块手册。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**configuration:** SNMP 配置告警信息。配置该参数后，系统会以 10 分钟为周期，查看周期内当前运行配置或者启动配置是否被修改，以及是否有用户对启动配置文件进行修改，并将最后一次修改形成一条告警输出。

**protocol:** 开启指定协议模块的 SNMP 告警功能。用户可通过执行 **snmp-agent trap enable ?** 命令来获取该参数的取值，有关此参数的详细介绍，请参见各模块的命令手册。

**standard:** SNMP 标准告警信息。包括以下五种：

- **authentication:** NMS 访问设备时认证失败，输出 SNMP 认证失败的告警信息。
- **coldstart:** 当设备重新启动时，输出设备冷启动告警信息。
- **linkdown:** 当接口的链路 down 时，输出 linkDown 告警信息。
- **linkup:** 当接口的链路 up 时，输出 linkUp 告警信息。
- **warmstart:** 当 SNMP 模块重新启动时，输出热启动告警信息。

**system:** SNMP 系统告警信息。配置该参数后，如果系统时间被修改、系统重启或系统主用启动软件包不可用，均会生成告警信息。

#### 【使用指导】

开启指定协议模块的告警功能后，该模块会生成告警信息，用于报告该模块的重要事件。生成的告警信息将发送到设备的 SNMP 模块，通过设置 SNMP 中告警信息的发送参数，来决定告警信息输出的相关属性。

不指定可选参数时，表示在全局下开启/关闭所有可选模块的告警功能。

#### 【举例】

# 开启 SNMP 认证失败的告警功能。

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

```
[Sysname] snmp-agent trap enable standard authentication
```

#### 【相关命令】

- **snmp-agent target-host**

### 1.1.33 snmp-agent trap if-mib link extended

**snmp-agent trap if-mib link extended** 命令用来对标准格式的 linkUp 或 linkDown 告警信息进行私有扩展。

**undo snmp-agent trap if-mib link extended** 命令用来恢复缺省情况。

#### 【命令】

```
snmp-agent trap if-mib link extended
undo snmp-agent trap if-mib link extended
```

#### 【缺省情况】

系统发送的 linkUp/linkDown 告警信息的格式为标准格式，不对其进行私有扩展。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【使用指导】

扩展格式的 linkUp/linkDown 告警信息由标准格式的 linkUp/linkDown 告警信息后增加接口描述和接口类型信息构成，使用扩展格式的告警信息有助于网络管理员快速定位问题。

需要注意的是，配置该命令后，设备发送的 linkUp/linkDown 告警信息为扩展格式的信息。如果 NMS 不支持扩展格式，可能会无法解析信息。

#### 【举例】

# 对标准格式的 linkUp/linkDown 告警信息进行私有扩展。

```
<Sysname> system-view
[Sysname] snmp-agent trap if-mib link extended
```

### 1.1.34 snmp-agent trap life

**snmp-agent trap life** 命令用来配置告警信息的保存时间。

**undo snmp-agent trap life** 命令用来恢复缺省情况。

#### 【命令】

```
snmp-agent trap life seconds
undo snmp-agent trap life
```

#### 【缺省情况】

SNMP 告警信息的保存时间为 120 秒。

#### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*seconds*: 超时时间，取值范围为 1~2592000，单位为秒。

### 【使用指导】

SNMP 模块使用队列来发送告警信息，告警信息进入消息发送队列时会启动一个存活定时器。如果直到定时器超时（即达到 **snmp-agent trap life** 命令配置的时间），告警信息还没有被发送出去，系统就会将该告警信息从发送队列中删除。

### 【举例】

```
# 配置告警信息的保存时间为 60 秒。
<Sysname> system-view
[Sysname] snmp-agent trap life 60
```

### 【相关命令】

- **snmp-agent trap enable**
- **snmp-agent target-host**
- **snmp-agent trap queue-size**

## 1.1.35 snmp-agent trap log

**snmp-agent trap log** 命令用来开启 SNMP 告警日志功能。

**undo snmp-agent trap log** 命令用来关闭 SNMP 告警日志功能。

### 【命令】

```
snmp-agent trap log
undo snmp-agent trap log
```

### 【缺省情况】

SNMP 告警日志功能处于关闭状态。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【使用指导】

打开 SNMP 告警日志开关，Agent 向 NMS 发送告警时，Agent 会向信息中心模块发送一条日志来记录该告警相关的信息。通过配置信息中心的参数，最终决定 SNMP 告警日志的输出规则（即是否允许输出以及输出方向）。

### 【举例】

```
# 开启 SNMP 告警日志功能。
<Sysname> system-view
[Sysname] snmp-agent trap log
```

### 1.1.36 snmp-agent trap queue-size

**snmp-agent trap queue-size** 命令用来配置告警信息发送队列的长度。

**undo snmp-agent trap queue-size** 命令用来恢复缺省情况。

#### 【命令】

```
snmp-agent trap queue-size size
undo snmp-agent trap queue-size
```

#### 【缺省情况】

告警信息的发送队列最多可以存储 100 条告警信息。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**size**: 消息队列中可以存储的告警信息的数目，取值范围 1~1000。

#### 【使用指导】

告警信息产生后，会进入告警信息消息队列进行发送，告警信息消息队列的长度决定了队列最多可以存储的告警信息的数目。当告警信息队列达到设定长度后，最新生成的告警信息会进入消息队列，最早产生的告警信息被丢弃。

#### 【举例】

# 配置发送告警信息消息队列最多可以存储 200 条告警信息。

```
<Sysname> system-view
[Sysname] snmp-agent trap queue-size 200
```

#### 【相关命令】

- **snmp-agent trap enable**
- **snmp-agent target-host**
- **snmp-agent trap life**

### 1.1.37 snmp-agent usm-user { v1 | v2c }

**snmp-agent usm-user { v1 | v2c }** 命令用来创建 SNMPv1 或 SNMPv2c 用户。

**undo snmp-agent usm-user { v1 | v2c }** 命令用来删除 SNMPv1 或 SNMPv2c 用户。

#### 【命令】

```
snmp-agent usm-user { v1 | v2c } user-name group-name [ acl { ipv4-acl-number |
name ipv4-acl-name } | acl ipv6 { ipv6-acl-number | name ipv6-acl-name } ] *
undo snmp-agent usm-user { v1 | v2c } user-name
```

#### 【缺省情况】

不存在 SNMP 用户。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**v1**: 表示配置的用户名适用于 SNMPv1 组网环境。

**v2c**: 表示配置的用户名适用于 SNMPv2c 组网环境。

**user-name**: 用户名，为 1~32 个字符的字符串，区分大小写。

**group-name**: 该用户对应组的名称，取值范围为 1~32 个字符的字符串，区分大小写。创建用户时，组可以不存在。但要使创建的用户生效，必须先创建组。

**acl**: 将用户名与基本/高级 IPv4 ACL 绑定。

**ipv4-acl-number**: 表示基本或高级 IPv4 ACL 的编号，其中基本 IPv4 ACL 的取值范围为 2000~2999，高级 IPv4 ACL 的取值范围为 3000~3999。

**name ipv4-acl-name**: 表示基本或高级 IPv4 ACL 的名称，为 1~63 个字符的字符串，不区分大小写。

**acl ipv6**: 将用户名与基本/高级 IPv6 ACL 绑定。

**ipv6-acl-number**: 表示基本或高级 IPv6 ACL 的编号，其中基本 IPv6 ACL 的取值范围为 2000~2999，高级 IPv6 ACL 的取值范围为 3000~3999。

**name ipv6-acl-name**: 表示基本或高级 IPv6 ACL 的名称，为 1~63 个字符的字符串，不区分大小写。

## 【使用指导】

FIPS 模式下，不支持本命令。

为了安全起见，只有具有 network-admin 或者 level-15 用户角色的用户登录设备后才能执行本命令。其它角色的用户，即使授权了 SNMP 特性或本命令的操作权限，也不能执行本命令。

SNMPv1 和 SNMPv2c 组网应用中 NMS 和 Agent 之间使用团体名来认证，SNMPv3 组网应用中使用用户名来认证。

用户有两种方式创建团体：

- 使用 **snmp-agent community** 命令来创建团体。
- 配置 **snmp-agent usm-user { v1 | v2c }** 和 **snmp-agent group { v1 | v2c }** 命令成功创建 SNMPv1 或 SNMPv2c 用户以及相应的组后，系统会以用户名为团体名自动创建一个团体。该团体的访问权限可通过 **snmp-agent community** 或 **snmp-agent group { v1 | v2c }** 命令来修改，最后一次执行的命令生效。

**display snmp-agent community** 会显示这两种方式创建的团体的信息。

创建 SNMP 组和用户的时候都可以使用 **acl** 参数限制非法 NMS 访问设备，只有两个 ACL 均允许的 NMS 才能访问设备。在创建组或用户时，ACL 均遵循以下规则：

- 当未引用 ACL、引用的 ACL 不存在、或者引用的 ACL 下没有配置规则时，允许所有 NMS 访问设备。
- 当引用的 ACL 下配置了规则时，则只有规则中 **permit** 的 NMS 才能访问设备，其他 NMS 不允许访问设备。

关于 ACL 的详细描述和介绍请参见“ACL 和 QoS 配置指导”中的“ACL”。

### 【举例】

# 在 SNMP 组 readCom 里创建 SNMPv2c 用户 userv2c。

```
<Sysname> system-view
[Sysname] snmp-agent sys-info version v2c
[Sysname] snmp-agent group v2c readCom
[Sysname] snmp-agent usm-user v2c userv2c readCom
```

如果 NMS 需要访问 Agent，则应将 NMS 的版本号指定为 SNMPv2c，Read community 选项填写为 userv2c。

# 在 SNMP 组 readCom 里创建 SNMPv2c 用户 userv2c，并且只允许 IP 地址为 1.1.1.1 的 NMS 使用该用户名访问 Agent，禁止其它 NMS 使用该用户名访问。

```
<Sysname> system-view
[Sysname] acl basic 2001
[Sysname-acl-ipv4-basic-2001] rule permit source 1.1.1.1 0.0.0.0
[Sysname-acl-ipv4-basic-2001] rule deny source any
[Sysname-acl-ipv4-basic-2001] quit
[Sysname] snmp-agent sys-info version v2c
[Sysname] snmp-agent group v2c readCom
[Sysname] snmp-agent usm-user v2c userv2c readCom acl 2001
```

# 在 SNMP 组 readCom 里创建 SNMPv2c 用户 userv2c，并且只允许 IP 地址为 1.1.1.2 的 NMS 使用该用户名访问 Agent，禁止其它 NMS 使用该用户名访问。

```
[Sysname] acl basic name testacl
[Sysname-acl-ipv4-basic-testacl] rule permit source 1.1.1.2 0.0.0.0
[Sysname-acl-ipv4-basic-testacl] rule deny source any
[Sysname-acl-ipv4-basic-testacl] quit
[Sysname] snmp-agent sys-info version v2c
[Sysname] snmp-agent group v2c readCom
[Sysname] snmp-agent usm-user v2c userv2c readCom acl name testacl
```

### 【相关命令】

- **snmp-agent group**
- **snmp-agent community**
- **display snmp-agent community**

## 1.1.38 snmp-agent usm-user v3

**snmp-agent usm-user v3** 命令用来创建 SNMPv3 用户。

**undo snmp-agent usm-user v3** 命令用来删除 SNMPv3 用户。

### 【命令】

非 FIPS 模式下：

- VACM 方式：  
**snmp-agent usm-user v3** *user-name* *group-name* [ **remote** { *ipv4-address* | *ipv6 ipv6-address* } ] [ { **cipher** | **simple** } **authentication-mode** { *md5* | *sha* } **auth-password** [ **privacy-mode** { *3des* | *aes128* | *aes192* | *aes256* | *des56* } ]

```
priv-password ] ] [ acl { ipv4-acl-number | name ipv4-acl-name } | acl ipv6
{ ipv6-acl-number | name ipv6-acl-name } ] *
```

```
undo snmp-agent usm-user v3 user-name { local | engineid engineid-string
| remote { ipv4-address | ipv6 ipv6-address } }
```

- RBAC 方式:

```
snmp-agent usm-user v3 user-name user-role role-name [ remote
{ ipv4-address | ipv6 ipv6-address } ] [ { cipher | simple }
authentication-mode { md5 | sha } auth-password [ privacy-mode { 3des |
aes128 | aes192 | aes256 | des56 } priv-password ] ] [ acl { ipv4-acl-number
| name ipv4-acl-name } | acl ipv6 { ipv6-acl-number | name
ipv6-acl-name } ] *
```

```
undo snmp-agent usm-user v3 user-name { local | engineid engineid-string
| remote { ipv4-address | ipv6 ipv6-address } }
```

FIPS 模式下:

- VACM 方式:

```
snmp-agent usm-user v3 user-name group-name [ remote { ipv4-address |
ipv6 ipv6-address } ] { cipher | simple } authentication-mode sha
auth-password [ privacy-mode { aes128 | aes192 | aes256 } priv-password ]
[ acl { ipv4-acl-number | name ipv4-acl-name } | acl ipv6
{ ipv6-acl-number | name ipv6-acl-name } ] *
```

```
undo snmp-agent usm-user v3 user-name { local | engineid engineid-string
| remote { ipv4-address | ipv6 ipv6-address } }
```

- RBAC 方式:

```
snmp-agent usm-user v3 user-name user-role role-name [ remote
{ ipv4-address | ipv6 ipv6-address } ] [ { cipher | simple }
authentication-mode sha auth-password [ privacy-mode { aes128 | aes192 |
aes256 } priv-password ] ] [ acl { ipv4-acl-number | name ipv4-acl-name }
| acl ipv6 { ipv6-acl-number | name ipv6-acl-name } ] *
```

```
undo snmp-agent usm-user v3 user-name { local | engineid engineid-string
| remote { ipv4-address | ipv6 ipv6-address } }
```

### 【缺省情况】

不存在 SNMPv3 用户。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*user-name*: 用户名, 为 1~32 个字符的字符串, 区分大小写。



**group-name**: 该用户对应组的名称, 取值范围为 1~32 个字符的字符串, 区分大小写。创建用户时, 组可以不存在。但要使创建的用户生效, 必须先创建组。

**user-role role-name**: 该用户对应的角色名称, *role-name* 为 1~63 个字符的字符串, 区分大小写。

**remote { ipv4-address | ipv6 ipv6-address }**: 接收 Inform 信息的目的主机的 IP 地址或者 IPv6 地址, 通常为 NMS 的 IP 地址或者 IPv6 地址。当设备需要向目的主机发送 SNMPv3 Inform 报文时, 该参数必须配置, 还需要使用 **snmp-agent remote** 命令将目的主机的 IP 地址或者 IPv6 地址和引擎 ID 绑定。

**cipher**: 以密文方式配置认证密码和加密密码。该密码将转换成对应的密文摘要存储在设备中。

**simple**: 以明文方式配置认证密码和加密密码。该密码将转换成对应的密文摘要存储在设备中。

**authentication-mode**: 指定认证算法。不指定该参数时, 表示不认证。取值为:

- **md5**: 采用 HMAC-MD5 算法。HMAC-MD5 的相关内容请参见“安全配置指导”中的“IPsec”。
- **sha**: 采用 HMAC-SHA1 算法。HMAC-SHA1 的相关内容请参见“安全配置指导”中的“IPsec”。

**auth-password**: 表示认证密码, 区分大小写, 具体要求如下:

- 以明文方式配置时, *auth-password* 表示明文认证密码: 非 FIPS 模式下, 认证密码的长度范围是 1~64 个字符; FIPS 模式下, 认证密码的长度范围是 15~64 字符, 密码元素的最少组合类型为 4 (必须包括数字、大写字母、小写字母以及特殊字符)。
- 以密文方式配置时, *auth-password* 表示密文认证密码, 该密码可通过 **snmp-agent calculate-password** 命令计算获得。

**privacy-mode**: 指定加密算法。不指定该参数时, 表示不加密。取值为:

- **3des**: 采用 3DES (Triple Data Encryption Standard, 三重数据加密标准) 算法, 密钥长度为 168 比特。
- **aes128**: 采用 AES (Advanced Encryption Standard, 高级加密标准) 算法, 密钥长度为 128 比特。
- **aes192**: 采用 AES 算法, 密钥长度为 192 比特。
- **aes256**: 采用 AES 算法, 密钥长度为 256 比特。
- **des56**: 采用 DES (Data Encryption Standard, 数据加密标准) 算法, 密钥长度为 56 比特。

**priv-password**: 表示加密密码, 区分大小写, 具体要求如下:

- 以明文方式配置时, *auth-password* 表示明文加密密码: 非 FIPS 模式下, 密码的长度范围是 1~64 个字符; FIPS 模式下, 加密密码的长度范围是 15~64 字符, 密码元素的最少组合类型为 4 (必须包括数字、大写字母、小写字母以及特殊字符)。
- 以密文方式配置时, *auth-password* 表示密文加密密码, 该密码可通过 **snmp-agent calculate-password** 命令计算获得。

**acl**: 将用户名与基本/高级 IPv4 ACL 绑定。

**ipv4-acl-number**: 表示基本或高级 IPv4 ACL 的编号, 其中基本 IPv4 ACL 的取值范围为 2000~2999, 高级 IPv4 ACL 的取值范围为 3000~3999。

**name ipv4-acl-name**: 表示基本或高级 IPv4 ACL 的名称, 为 1~63 个字符的字符串, 不区分大小写。

**acl ipv6**: 将用户名与基本/高级 IPv6 ACL 绑定。

*ipv6-acl-number*: 表示基本或高级 IPv6 ACL 的编号, 其中基本 IPv6 ACL 的取值范围为 2000~2999, 高级 IPv6 ACL 的取值范围为 3000~3999。

**name** *ipv6-acl-name*: 表示基本或高级 IPv6 ACL 的名称, 为 1~63 个字符的字符串, 不区分大小写。

**local**: 表示本地实体引擎 ID。缺省情况下, 创建的 SNMPv3 用户与本地 SNMP 实体引擎相关联。

**engineid** *engineid-string*: 指定与该用户相关联的引擎 ID 字符串, 必须为偶数个十六进制数, 偶数的取值范围为 10~64, 不区分大小写。全 0 和全 F 均被认为是无效参数。由于 SNMPv3 版本的用户名、密文密码等都和引擎 ID 相关联, 如果更改了引擎 ID, 则原引擎 ID 下配置的用户名、密码失效, 更改后可以使用该参数将 *engineid-string* 指定为创建该用户时的本地引擎 ID 来删除失效用户名。

### 【使用指导】

为了安全起见, 只有具有 **network-admin** 或者 **level-15** 用户角色的用户登录设备后才能执行本命令。其它角色的用户, 即使授权了 SNMP 特性或本命令的操作权限, 也不能执行本命令。

创建 SNMPv3 用户时, 可以通过两种配置方式来控制用户访问的权限:

- 通过 VACM 方式配置的 SNMP 用户依附于 SNMP 组, 创建用户时, 请先创建组。否则, 用户能够创建成功但是不生效。一个组可以包含多个用户。组定义了用户能够访问的 SNMP 对象 (通过 MIB 视图来限定) 以及是否进行认证和加密等, 而认证和加密的具体算法和密码则是在创建用户时定义。
- 通过 RBAC 方式配置的 SNMP 用户依附于用户角色, 创建用户时, 通过 **user-role role-name** 参数配置用户的角色。用户角色定义了 SNMP 用户能够访问的 SNMP 对象以及操作类型 (通过 **rule** 规则来限定)。使用 RBAC 方式创建 SNMP v3 用户后, 还可以使用 **snmp-agent usm-user v3 user-role** 命令为该用户绑定更多的用户角色, 最多可绑定 64 个用户角色。

推荐使用 RBAC 配置方式, 安全性更高。

通过 VACM 和 RBAC 方式配置 SNMP 用户时, 需要注意:

- 通过 VACM 方式配置 SNMP 用户时, 当用户名相同, 多次执行本命令, 最后一次执行的命令生效。
- 通过 RBAC 方式配置 SNMP 用户时, 可以多次使用本命令为已创建的 SNMPv3 用户添加角色, 若未配置其他参数, 则其他配置不变, 只添加角色; 若同时配置其他参数 (如认证方式), 则为用户添加角色, 同时修改其他配置。

创建 SNMP 组和用户的时候都可以使用 **acl** 参数限制非法 NMS 访问设备, 只有两个 ACL 均允许的 NMS 才能访问设备。在创建组或用户时, ACL 均遵循以下规则:

- 当未引用 ACL、引用的 ACL 不存在、或者引用的 ACL 下没有配置规则时, 允许所有 NMS 访问设备。
- 当引用的 ACL 下配置了规则时, 则只有规则中 **permit** 的 NMS 才能访问设备, 其他 NMS 不允许访问设备。

关于 ACL 的详细描述和介绍请参见“ACL 和 QoS 配置指导”中的“ACL”。

### 【举例】

# VACM 方式: 为 v3 组 **testGroup** 加入一个用户 **testUser**, 安全级别为只认证不加密, 认证算法为 HMAC-SHA1, 认证密码为明文 **123456TESTplat&!**。

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

```
[Sysname] snmp-agent group v3 testGroup authentication
```

```
[Sysname] snmp-agent usm-user v3 testUser testGroup simple authentication-mode sha 123456TESTplat&!
```

在 NMS 上配置：版本号为 SNMPv3，用户名为 testUser，认证算法为 HMAC-SHA1，认证密码为 123456TESTplat&！，建立连接，就可以对设备上缺省视图内的 MIB 对象进行访问了。

# VACM 方式：为 v3 组 testGroup 加入一个用户 testUser，安全级别为认证和加密，认证算法为 HMAC-SHA1、加密算法为 AES，认证密码为明文 123456TESTauth&！，加密密码为明文 123456TESTencr&！。

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

```
[Sysname] snmp-agent group v3 testGroup privacy
```

```
[Sysname] snmp-agent usm-user v3 testUser testGroup simple authentication-mode sha 123456TESTauth&! privacy-mode aes128 123456TESTencr&!
```

在 NMS 上配置：版本号为 SNMPv3，用户名为 testUser，认证算法为 HMAC-SHA1，认证密码为 123456TESTauth&！，加密算法为 AES，加密密码为 123456TESTencr&！，建立连接，就可以对设备上缺省视图内的 MIB 对象进行访问了。

# VACM 方式：为 v3 组 testGroup 加入一个与 IP 为 10.1.1.1 的远端 SNMP 实体引擎相关联的 SNMPv3 用户 remoteUser，安全级别为认证和加密，认证算法为 HMAC-SHA1、加密算法为 AES，认证密码为明文 123456TESTauth&！，加密密码为明文 123456TESTencr&！。

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

```
[Sysname] snmp-agent remote 10.1.1.1 engineid 123456789A
```

```
[Sysname] snmp-agent group v3 testGroup privacy
```

```
[Sysname] snmp-agent usm-user v3 remoteUser testGroup remote 10.1.1.1 simple authentication-mode sha 123456TESTauth&! privacy-mode aes128 123456TESTencr&!
```

# RBAC 方式：创建一个新的 SNMPv3 用户 testUser，角色为 network-operator，安全级别为只认证不加密，认证算法为 HMAC-SHA1，认证密码为明文 123456TESTplat&！。

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

```
[Sysname] snmp-agent usm-user v3 testUser user-role network-operator simple authentication-mode sha 123456TESTplat&!
```

在 NMS 上配置：版本号为 SNMPv3，用户名为 testUser，认证算法为 HMAC-SHA1，认证密码为 123456TESTplat&！，建立连接，就可以对设备上所有 MIB 对象进行只读操作。

## 【相关命令】

- **display snmp-agent usm-user**
- **snmp-agent group**
- **snmp-agent calculate-password**
- **snmp-agent remote**
- **snmp-agent usm-user v3 user-role**

### 1.1.39 snmp-agent usm-user v3 user-role

**snmp-agent usm-user v3 user-role** 命令用来为通过 RBAC 方式创建的 SNMPv3 用户添加角色。

**undo snmp-agent usm-user user-role** 命令用来为 SNMPv3 用户删除角色。

### 【命令】

```
snmp-agent usm-user v3 user-name user-role role-name  
undo snmp-agent usm-user v3 user-name user-role role-name
```

### 【缺省情况】

使用创建 SNMPv3 用户时指定的角色。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**user-name**: 用户名，为 1~32 个字符的字符串，区分大小写。

**user-role role-name**: 该用户对应的角色名称，**role-name** 为 1~63 个字符的字符串，区分大小写。

### 【使用指导】

一个 RBAC 方式配置的 SNMPv3 用户可配置多个用户角色。用户可以通过本命令来为通过 RBAC 方式创建的 SNMPv3 用户添加与删除角色，最多可以配置 64 个有效的用户角色且至少保留一个用户角色。

### 【举例】

# 已创建 SNMPv3 用户 testUser 拥有 network-operato 用户角色，现为用户 testUser 添加 network-admin 用户角色。

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

```
[Sysname] snmp-agent usm-user v3 testUser user-role network-admin
```

### 【相关命令】

- **snmp-agent usm-user v3**

# 目 录

1 RMON.....	1-1
1.1 RMON配置命令 .....	1-1
1.1.1 display rmon alarm.....	1-1
1.1.2 display rmon event.....	1-2
1.1.3 display rmon eventlog.....	1-3
1.1.4 display rmon history .....	1-5
1.1.5 display rmon prialarm .....	1-6
1.1.6 display rmon statistics.....	1-8
1.1.7 rmon alarm.....	1-10
1.1.8 rmon event.....	1-11
1.1.9 rmon history .....	1-12
1.1.10 rmon prialarm.....	1-13
1.1.11 rmon statistics .....	1-15

# 1 RMON

## 1.1 RMON配置命令

### 1.1.1 display rmon alarm

`display rmon alarm` 命令用来显示 RMON 告警表项的相关信息。

【命令】

`display rmon alarm [ entry-number ]`

【视图】

任意视图

【缺省用户角色】

network-admin  
network-operator

【参数】

*entry-number*: 告警表项的索引号，取值范围为 1~65535。如果不指定索引号，则显示所有告警表项的相关信息。

【举例】

# 显示所有 RMON 告警表项的相关信息。

```
<Sysname> display rmon alarm
AlarmEntry 1 owned by user1 is VALID.
Sample type                : absolute
Sampled variable           : 1.3.6.1.2.1.16.1.1.1.4.1<etherStatsOctets.1>
Sampling interval (in seconds) : 10
Rising threshold           : 50(associated with event 1)
Falling threshold          : 5(associated with event 2)
Alarm sent upon entry startup : risingOrFallingAlarm
Latest value                : 0
```

表1-1 display rmon alarm 命令显示信息描述表

字段	描述
AlarmEntry <i>entry-number</i> owned by <i>owner</i> is <i>status</i> .	<i>owner</i> 创建的告警表项 <i>entry-number</i> 的当前状态为 <i>status</i> <ul style="list-style-type: none"><li><i>entry-number</i>: 告警表项</li><li><i>owner</i>: 该表项创建者</li><li><i>status</i>: 与该索引对应的告警表项的状态（VALID 表示有效，UNDERCREATION 表示无效。处于无效状态的表项使用 <code>display rmon alarm</code> 命令可以查看到，但使用 <code>display current-configuration</code> 和 <code>display this</code> 看不到对应的 <code>rmon alarm</code> 配置命令）。命令行配置告警表项时不可配且缺省为 VALID</li></ul>

字段	描述
Sample type	采样类型，取值为： <ul style="list-style-type: none"> <li><b>absolute</b>: 绝对值采样</li> <li><b>delta</b>: 变化值采样</li> </ul>
Sampled variable	告警变量，即被监控的MIB节点
Sampling interval	采样的时间间隔，单位为秒
Rising threshold	上限阈值（当采样值大于等于该值时引发上限告警）
associated with event	告警对应的事件索引
Falling threshold	下限阈值（当采样值小于等于该值时引发下限告警）
Alarm sent upon entry startup	初次触发告警类型： <ul style="list-style-type: none"> <li><b>risingAlarm</b>: 表示触发上限告警</li> <li><b>fallingAlarm</b>: 表示触发下限告警</li> <li><b>risingorFallingAlarm</b>: 表示触发上限或下限告警</li> </ul> 缺省情况下，触发risingorFallingAlarm类型告警
Latest value	最近一次采样值

#### 【相关命令】

- rmon alarm**

#### 1.1.2 display rmon event

**display rmon event** 命令用来显示 RMON 事件表项相关信息。

#### 【命令】

**display rmon event** [*entry-number*]

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

**entry-number**: 事件表项的索引号，取值范围为 1~65535。如果不指定索引号，则显示所有事件表项的相关信息。

#### 【使用指导】

显示信息包括：事件表中的事件索引、事件的所有者、对事件的描述、事件引发的动作（日志或告警信息）、最近一次事件发生的时刻（此时间是以系统初始化/启动以来的秒数计算的）等。

#### 【举例】

# 显示所有 RMON 事件表项相关信息。

```
<Sysname> display rmon event
EventEntry 1 owned by user1 is VALID.
  Description: N/A
  Community: Security
  Take the action log-trap when triggered, last triggered at 0days 00h:02m:27s uptime.
```

表1-2 display rmon event 命令显示信息描述表

字段	描述
EventEntry entry-number owned by owner is status.	Owner创建的事件表项entry-number的当前状态为status <ul style="list-style-type: none"><li>entry-number: 事件表项</li><li>owner: 该表项创建者</li><li>status: 与该索引对应的事件表项的状态（VALID 表示有效，UNDERCREATION 表示无效。处于无效状态的表项使用 <b>display rmon event</b> 命令可以查看到，但使用 <b>display current-configuration</b> 和 <b>display this</b> 看不到对应的 <b>rmon event</b> 配置命令）。命令行配置 event 表项时不可配且默认为 VALID</li></ul>
Description	该事件表项的描述
Community	接收告警信息的网管站的团体名
Take the action action when triggered	事件触发时采取的动作： <ul style="list-style-type: none"><li>none: 表示不采取任何措施</li><li>log: 表示事件被触发时会记录日志</li><li>trap: 表示事件被触发时会生成告警信息发送给设备的 SNMP 模块</li><li>log-trap: 表示事件被触发时既会记录日志又会生成告警信息发送给设备的 SNMP 模块</li></ul>
last triggered at time uptime	最近一次事件发生的时间（设备启动以来的时间）

【相关命令】

- rmon event

1.1.3 display rmon eventlog

display rmon eventlog 命令用来显示事件日志表项的相关信息。

【命令】

```
display rmon eventlog [ entry-number ]
```

【视图】

任意视图

【缺省用户角色】

network-admin  
network-operator

【参数】

entry-number: 事件表项的索引号，取值范围为 1~65535。如果不指定索引号，则显示所有事件的日志表项的相关信息。



【使用指导】

如果使用 **rmon event** 命令指定某表项的动作包括记录日志，当该事件被触发时，就会在 **RMON** 事件日志表中记录一条该事件的日志。通过该命令可以显示事件日志表的具体内容：事件表中的事件索引及事件当前的状态、事件产生日志的时刻（此时间是以系统初始化/启动以来的秒数计算的）以及事件的描述等。

每个事件最多有 10 条日志。超过 10 条日志后，新日志将覆盖最早的日志。

【举例】

# 查看由告警表产生的 **RMON** 事件 99 的日志。

```
<Sysname> display rmon eventlog 99
EventEntry 99 owned by ww is VALID.
  LogEntry 99.1 created at 50days 08h:54m:44s uptime.
  Description: The 1.3.6.1.2.1.16.1.1.1.4.5 defined in alarmEntry 77,
    uprise 16760000 with alarm value 16776314. Alarm sample type is absolute.
  LogEntry 99.2 created at 50days 09h:11m:13s uptime.
  Description: The 1.3.6.1.2.1.16.1.1.1.4.5 defined in alarmEntry 77,
    less than(or =) 20000000 with alarm value 16951648. Alarm sample type is absolute.
  LogEntry 99.3 created at 50days 09h:18m:43s uptime.
  Description: The alarm formula defined in prialarmEntry 777,
    less than(or =) 15000000 with alarm value 14026493. Alarm sample type is absolute.
  LogEntry 99.4 created at 50days 09h:23m:28s uptime.
  Description: The alarm formula defined in prialarmEntry 777,
    uprise 17000000 with alarm value 17077846. Alarm sample type is absolute.
```

以上举例表明事件 99 产生的告警事件日志，其中告警表产生的事件日志 2 条，扩展告警表产生的事件日志 2 条：

- 日志 99.1 由告警表项 77 触发生成，原因是告警值(16776314)超过了上限阈值(16760000)，采样类型为绝对值采样。
- 日志 99.2 由告警表项 77 触发生成，原因是告警值(16951648)低于下限阈值(20000000)，采样类型为绝对值采样。
- 日志 99.3 由扩展告警表项 777 触发生成，原因是告警值(14026493)低于下限阈值(15000000)，采样类型为绝对值采样。
- 日志 99.4 由扩展告警表项 777 触发生成，原因是告警值(17077846)超过上限阈值(17000000)，采样类型为绝对值采样。

表1-3 display rmon eventlog 命令显示信息描述表

字段	描述
EventEntry entry-number owned by owner is status.	<p>Owner创建的事件日志表项entry-number的当前状态为status</p> <ul style="list-style-type: none"><li>• entry-number: 事件日志表项</li><li>• owner: 该表项创建者</li><li>• status: 与该索引对应的事件日志表项的状态（VALID 表示有效，UNDERCREATION 表示无效。处于无效状态的表项使用 <b>display rmon eventlog</b> 可以查看到，但使用 <b>display current-configuration</b> 和 <b>display this</b> 看不到对应的 <b>rmon eventlog</b> 配置命令）。命令行配置 event 表项时不可配且默认为 VALID</li></ul>

字段	描述
LogEntry <i>entry-number</i> created at <i>created-time</i> uptime.	日志表项 <i>entry-number</i> 的创建时间为 <i>created-time</i> <ul style="list-style-type: none"> <li><i>entry-number</i>: 日志表项索引号, 表示方式为 logEventIndex.logIndex, logEventIndex 和 logIndex 为 MIB 节点</li> <li><i>created-time</i>: 日志表项的创建时间</li> </ul>
Description	该条日志的描述

## 【相关命令】

- `rmon event`

### 1.1.4 display rmon history

`display rmon history` 命令用来显示 RMON 历史控制表及历史采样信息。

## 【命令】

`display rmon history [ interface-type interface-number ]`

## 【视图】

任意视图

## 【缺省用户角色】

network-admin  
network-operator

## 【参数】

*interface-type interface-number*: 指定接口类型和接口编号。如果未指定本参数, 则显示所有接口下配置的历史控制表及历史采样信息。

## 【使用指导】

在端口创建历史表项之后, 系统会按一定的时间周期统计端口的信息, 并将这些信息保存到 etherHistoryEntry 表, 使用本命令可以显示该表项存储的记录。

可显示的历史采样信息的数目以及历史采样的间隔可以通过 `rmon history` 命令来设置。

## 【举例】

# 显示端口 GigabitEthernet1/0/1 的 RMON 历史控制表及历史采样信息。

```
<Sysname> display rmon history gigabitethernet 1/0/1
HistoryControlEntry 6 owned by user1 is VALID.
  Sampled interface      : GigabitEthernet1/0/1<ifIndex.117>
  Sampling interval      : 8(sec) with 3 buckets max
  Sampling record 1 :
    dropevents           : 0           , octets           : 5869
    packets              : 54          , broadcast packets : 9
    multicast packets    : 23          , CRC alignment errors : 0
    undersize packets    : 0           , oversize packets   : 0
    fragments            : 0           , jabbers            : 0
    collisions           : 0           , utilization        : 0
```

表1-4 display rmon history 命令显示信息描述表

字段	描述
HistoryControlEntry entry-number owned by owner is status.	<p>Owner创建的历史控制表项entry-number的当前状态为status</p> <ul style="list-style-type: none"> <li>entry-number: 历史控制表项</li> <li>owner: 该表项的创建者</li> <li>status: 与该索引对应的历史控制表项的状态（VALID 表示有效，UNDERCREATION 表示无效。处于无效状态的表项使用 <b>display rmon history</b> 命令可以查看到，但使用 <b>display current-configuration</b> 和 <b>display this</b> 看不到对应的 <b>rmon history</b> 配置命令）。命令行配置 HistoryControl 表项时不可配置且默认为 VALID</li> </ul>
Sampled Interface	被统计的接口
Sampling interval	统计周期，单位为秒，系统会按周期对端口的信息进行统计
buckets max	<p>系统最多可保存的统计值的条数</p> <ul style="list-style-type: none"> <li>如果在 <b>rmon history</b> 命令中指定的 <b>buckets</b> 的值超出了设备实际支持的历史表容量，则此处显示的是设备实际支持的历史表容量</li> <li>如果当前保存的统计值条数已经到达了系统支持的最大值，则系统会删除最早的记录来保存新的统计值</li> </ul>
Sampling record	历史采样表项索引号
dropevents	统计周期内检测到的丢包事件次数
octets	统计周期内接收到的字节数
packets	统计周期内接收到的包数
broadcast packets	统计周期内接收到的广播包数
multicast packets	统计周期内接收到的组播包数
CRC alignment errors	统计周期内接收到的校验错误的包数
undersize packets	<p>统计周期内接收到的过小的包数</p> <p>过小包指的是长度小于64字节的包（不包括帧标志位，包括FCS校验码）</p>
oversize packets	<p>统计周期内接收到的超大的包数</p> <p>超大包指的是长度大于1518字节的包（不包括帧标志位，包括FCS校验码）</p>
fragments	统计周期内接收到的过小且校验错误的包数
jabbers	统计周期内接收到的超大且校验错误的包数
collisions	统计周期内接收到的冲突的包数
utilization	统计周期内的带宽利用率（用百分比表示）

## 【相关命令】

- rmon history**

### 1.1.5 display rmon prialarm

**display rmon prialarm** 命令用来显示扩展告警表项的相关信息。

【命令】

`display rmon prialarm [ entry-number ]`

【视图】

任意视图

【缺省用户角色】

network-admin  
network-operator

【参数】

*entry-number*: 扩展告警表项的索引，取值范围为 1~65535。如果不指定索引号，则显示所有扩展告警表项的相关信息。

【举例】

# 显示 RMON 所有的扩展告警表项的相关信息。

```
<Sysname> display rmon prialarm
PrialarmEntry 1 owned by user1 is VALID.
Sample type                : absolute
Variable formula            : (.1.3.6.1.2.1.16.1.1.1.6.1*100/.1.3.6.1.2.1.16.1.1.1.5.1)
Description                 : ifUtilization.GigabitEthernet1/0/1
Sampling interval (in seconds) : 10
Rising threshold            : 80(associated with event 1)
Falling threshold           : 5(associated with event 2)
Alarm sent upon entry startup : risingOrFallingAlarm
Entry lifetime              : forever
Latest value                : 85
```

表1-5 display rmon prialarm 命令显示信息描述表

字段	描述
PrialarmEntry <i>entry-number</i> owned by <i>owner</i> is <i>status</i> .	<i>Owner</i> 创建的扩展告警表项 <i>entry-number</i> 的当前状态为 <i>status</i> <ul style="list-style-type: none"><li><i>entry-number</i>: 扩展告警表项</li><li><i>owner</i>: 该表项创建者</li><li><i>status</i>: 与该索引对应的扩展告警表项的状态（VALID 表示有效，UNDERCREATION 表示无效。处于无效状态的表项使用 <b>display rmon prialarm</b> 命令可以查看到，但使用 <b>display current-configuration</b> 和 <b>display this</b> 看不到相应的 <b>rmon prialarm</b> 配置命令）。命令行配置 <b>prialarm</b> 表项时不可配且默认为 VALID</li></ul>
Sample type	采样类型，取值为： <ul style="list-style-type: none"><li><b>absolute</b>: 绝对值采样</li><li><b>delta</b>: 变化值采样</li></ul>
Variable formula	样本变量的计算公式
Description	扩展告警表项的描述信息
Sampling interval	采样间隔，单位为秒，系统会按一定的时间间隔对采样变量进行绝对值采样或者变化值采样

字段	描述
Rising threshold	告警上限，当采样值大于等于该值时引发上限告警
Falling threshold	告警下限，当采样值小于等于该值时引发下限告警
associated with event	告警对应的事件索引
Alarm sent upon entry startup	初次触发告警类型： <ul style="list-style-type: none"> <li>risingAlarm：表示触发上限告警</li> <li>fallingAlarm：表示触发下限告警</li> <li>risingorFallingAlarm：表示触发上限或下限告警</li> </ul> 缺省情况下，触发risingorFallingAlarm类型告警
Entry lifetime	该扩展告警表项的存活时间，可以是永远存在，也可以是在规定的时间内存在
Latest value	最近一次采样值

### 【相关命令】

- `rmon prialarm`

### 1.1.6 display rmon statistics

`display rmon statistics` 命令用来显示 RMON 统计信息。

### 【命令】

`display rmon statistics [ interface-type interface-number ]`

### 【视图】

任意视图

### 【缺省用户角色】

network-admin  
network-operator

### 【参数】

*interface-type interface-number*：指定接口类型和接口编号。如果未指定本参数，则显示所有接口下配置的统计表及统计信息。

### 【使用指导】

本命令显示的是从端口创建统计表项到执行显示命令这段时间内端口的统计信息。设备重启时，会清除该统计信息。

### 【举例】

# 显示以太网接口 GigabitEthernet1/0/1 的 RMON 统计信息。

```
<Sysname> display rmon statistics gigabitethernet 1/0/1
EtherStatsEntry 1 owned by user1 is VALID.
  Interface : GigabitEthernet1/0/1<ifIndex.3>
  etherStatsOctets      : 43393306   , etherStatsPkts      : 619825
  etherStatsBroadcastPkts : 503581   , etherStatsMulticastPkts : 44013
```

```

etherStatsUndersizePkts : 0          , etherStatsOversizePkts : 0
etherStatsFragments    : 0          , etherStatsJabbers    : 0
etherStatsCRCAlignErrors : 0        , etherStatsCollisions : 0
etherStatsDropEvents (insufficient resources): 0
Incoming packets by size:
64      : 0          , 65-127 : 0          , 128-255 : 0
256-511: 0          , 512-1023: 0        , 1024-1518: 0

```

表1-6 display rmon statistics 命令显示信息描述表

字段	描述
EtherStatsEntry <i>entry-number</i> owned by <i>owner</i> is <i>status</i> .	<p><i>Owner</i>创建的统计信息表项<i>entry-number</i>的当前状态为<i>status</i></p> <ul style="list-style-type: none"> <li><b>entry-number:</b> 统计信息表项</li> <li><b>owner:</b> 该表项创建者</li> <li><b>status:</b> 与该索引对应的统计表项的状态（VALID 表示有效，UNDERCREATION 表示无效。处于无效状态的表项使用 <b>display rmon statistics</b> 命令可以查看到，但使用 <b>display current-configuration</b> 和 <b>display this</b> 看不到对应的 <b>rmon statistics</b> 配置命令）。命令行配置 statistics 表项时不可配且默认为 VALID</li> </ul>
Interface	被统计端口
etherStatsOctets	统计时间内，端口收到的所有报文的字节数
etherStatsPkts	统计时间内，端口收到的所有报文的包数
etherStatsBroadcastPkts	统计时间内，端口收到的所有广播包的数量
etherStatsMulticastPkts	统计时间内，端口收到的所有组播包的数量
etherStatsUndersizePkts	统计时间内，端口收到的所有过小包的数量
etherStatsOversizePkts	统计时间内，端口收到的所有超大包的数量
etherStatsFragments	统计时间内，端口收到的所有过小且校验错误包的数量
etherStatsJabbers	统计时间内，端口收到的所有超大且校验错误包的数量
etherStatsCRCAlignErrors	统计时间内，端口收到的所有校验错误包的数量
etherStatsCollisions	统计时间内，端口收到的所有冲突包的数量
etherStatsDropEvents	统计时间内，端口收到的所有丢包事件的数量
Incoming packets by size: 64: 65-127: 128-255: 256-511: 512-1023: 1024-1518:	<p>统计时间内，根据包的长度对接收到的包分区间进行统计。包括：</p> <ul style="list-style-type: none"> <li>长度等于 64 字节的报文的个数</li> <li>长度为 65~127 字节的报文的个数</li> <li>长度为 128~255 字节的报文的个数</li> <li>长度为 256~511 字节的报文的个数</li> <li>长度为 512~1023 字节的报文的个数</li> <li>长度为 1024~1518 字节的报文的个数</li> </ul>

## 【相关命令】

- rmon statistics**

### 1.1.7 rmon alarm

**rmon alarm** 命令用来创建告警表项。

**undo rmon alarm** 命令用来删除指定的告警表项。

#### 【命令】

```
rmon alarm entry-number alarm-variable sampling-interval { absolute | delta }  
[ startup-alarm { falling | rising | rising-falling } ] rising-threshold  
threshold-value1 event-entry1 falling-threshold threshold-value2  
event-entry2 [ owner text ]  
undo rmon alarm entry-number
```

#### 【缺省情况】

告警表中不存在表项。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**entry-number**: 告警表项的索引号，取值范围为 1~65535。

**alarm-variable**: 告警变量，为 1~255 个字符的字符串，可以是节点 OID 的点分格式（格式为 *entry.integer.instance* 或者 *叶子节点名.instance*，如 1.3.6.1.2.1.2.1.10.1），也可以是节点名（如 *ifInOctets.1*）。只有可以解析为 ASN.1 中 INTEGER（INTEGER, INTEGER32, Unsigned32, Counter32, Counter64, Gauge, or TimeTicks）的数据类型的变量能作为告警变量，比如 *etherStatsEntry* 表项的叶子节点（*etherStatsOctets*、*etherStatsPkts* 和 *etherStatsBroadcastPkts* 等）的实例，*ifEntry* 表项的叶子节点（*ifInOctets*、*ifInUcastPkts* 和 *ifInNUcastPkts* 等）的实例。

**sampling-interval**: 采样间隔时间，取值范围为 5~65535，单位为秒。

**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。

**startup-alarm**: 表示初次采样时，如果达到或超出阈值，触发的告警类型。如果未指定本参数，触发 **rising-falling** 类型告警。

**rising**: 表示只触发上限告警。

**falling**: 表示只触发下限告警。

**rising-falling**: 表示触发上限或下限告警。

**rising-threshold threshold-value1 event-entry1**: 设置上限参数，*threshold-value1* 表示上限阈值，取值范围为-2147483648~2147483647；*event-entry1* 表示上限阈值相应的事件索引号，取值范围为 0~65535（0 表示没有对应的事件，告警被触发后不会采取任何事件动作）。

**falling-threshold threshold-value2 event-entry2**: 设置下限参数，*threshold-value2* 表示下限阈值，取值范围为-2147483648~2147483647；*event-entry2* 表示下限阈值相应的事件索引号，取值范围为 0~65535（0 表示没有对应的事件，告警被触发后不会采取任何事件动作）。

**owner text**: 该表项的创建者，为 1~127 个字符的字符串，区分大小写。

### 【使用指导】

本命令用来设置告警项，以便在出现异常时触发告警事件，再由告警事件来定义具体的处理方式。用户定义了告警表项后，系统会按照定义的时间周期去获取被监视的告警变量的值，并将该值和设定的阈值进行比较，去执行相应的处理过程。当采样值大于等于设定的上限 *threshold-value1*，触发事件表中定义的事件 *event-entry1*；采样值小于等于设定的下限 *threshold-value2*，触发事件表中定义的事件 *event-entry2*。

在添加告警表项之前，需要通过 **rmon event** 命令定义好告警表项中引用的事件。否则，虽然会创建告警表项，但是不能触发告警事件。

如果在新建表项时，指定的告警变量 (*alarm-variable*)、采样间隔 (*sampling-interval*)、采样类型 (**absolute** 或 **delta**)、上限阈值 (*threshold-value1*) 和下限阈值 (*threshold-value2*) 五项参数的值和已经存在的告警表项对应的五项参数值完全相同时，系统将认为这两个表项的配置相同，创建操作失败。

用户最多可以定义 60 个告警表项。

### 【举例】

# 在告警表中添加表项 1，对节点 1.3.6.1.2.1.16.1.1.1.4.1 以 10 秒的采样间隔进行绝对值采样，当采样值大于等于 5000 的上限阈值触发事件 1，小于等于下限阈值 5 时触发事件 2，创建者为 user1。

```
<Sysname> system-view
[Sysname] rmon event 1 log
[Sysname] rmon event 2 none
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] rmon statistics 1
[Sysname-GigabitEthernet1/0/1] quit
[Sysname] rmon alarm 1 1.3.6.1.2.1.16.1.1.1.4.1 10 absolute rising-threshold 5000 1
falling-threshold 5 2 owner user1
```

1.3.6.1.2.1.16.1.1.1.4 是叶子节点 **etherStatsOctets** 的 OID，它表示接口收到报文的统计值（以字节为单位）。以上步骤中也可以使用 **etherStatsOctets.1** 来代替 1.3.6.1.2.1.16.1.1.1.4.1 参数，.1 与接口统计表项的编号一致，如果创建的是“rmon statistics 5”，则对应需要使用 **etherStatsOctets.5**。

### 【相关命令】

- **display rmon alarm**
- **rmon event**

## 1.1.8 rmon event

**rmon event** 命令用来创建事件表项。

**undo rmon event** 命令用来删除指定的事件表项。

### 【命令】

```
rmon event entry-number [ description string ] { log | log-trap  
security-string | none | trap security-string } [ owner text ]  
undo rmon event entry-number
```



### 【缺省情况】

事件表中不存在表项。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**entry-number**: 事件表项的索引号，取值范围为 1~65535。

**description string**: 事件的描述信息，为 1~127 个字符的字符串，区分大小写。

**log**: 日志事件。当该事件被触发时，系统会记录日志。

**log-trap**: 日志和告警事件。当该事件被触发时，系统会同时记录日志和生成告警信息，生成的告警信息将发送到设备的 **SNMP** 模块。通过设置 **SNMP** 中告警信息的发送参数，来决定告警信息输出的相关属性。有关告警信息的详细介绍，请参见“网络管理和监控配置指导”中的“**SNMP**”。

**security-string**: 表示 **RMON** 生成的告警信息中携带的团体名。为 1~127 个字符的字符串，区分大小写。此处支持配置，但配置不生效。告警信息中携带的团体名由 **SNMP** 模块的配置决定。

**none**: 不产生动作的事件。当该事件被触发时，系统不做处理。

**trap**: 告警事件。当该事件被触发时，生成告警信息，生成的告警信息将发送到设备的 **SNMP** 模块。通过设置 **SNMP** 中告警信息的发送参数，来决定告警信息输出的相关属性。有关告警信息的详细介绍，请参见“网络管理和监控配置指导”中的“**SNMP**”。

**owner text**: 该表项的创建者，为 1~127 个字符的字符串，区分大小写。

### 【使用指导】

**RMON** 的事件管理定义事件索引号及事件的处理方式包括：记录日志、生成告警信息发送给设备的 **SNMP** 模块、记录日志的同时生成告警信息发送给设备的 **SNMP** 模块或者既不记录日志也生成告警信息发送给设备的 **SNMP** 模块。这样系统就可以对告警表中定义的告警事件进行相应的处理。事件组中定义的事件索引号对应告警组中指定事件索引号。

用户最多可以定义 60 个事件表项。

### 【举例】

# 在事件表中添加索引号为 10、类型为日志的事件，创建者为 user1。

```
<Sysname> system-view
[Sysname] rmon event 10 log owner user1
```

### 【相关命令】

- **display rmon event**
- **rmon alarm**
- **rmon prialarm**

## 1.1.9 rmon history

**rmon history** 命令用来创建历史控制表项。

**undo rmon history** 命令用来删除指定的历史表项。

## 【命令】

```
rmon history entry-number buckets number interval interval [ owner text ]
undo rmon history entry-number
```

## 【缺省情况】

历史控制表中不存在表项。

## 【视图】

以太网接口视图

## 【缺省用户角色】

network-admin

## 【参数】

**entry-number**: 历史控制表项的索引号，取值范围为 1~65535。

**buckets number**: 该历史控制表项对应的历史表容量，即历史表最多可容纳的记录数，取值范围为 1~65535，但实际配置如果超过 50 时，会提示取 50 最大配置值。

**interval interval**: 统计周期，取值范围为 5~3600，单位为秒。

**owner text**: 该表项的创建者，为 1~127 个字符的字符串，区分大小写。

## 【使用指导】

创建历史控制表项后，系统会按周期统计当前端口收发报文的情况，并将统计值作为一个实例保存在 **etherHistoryEntry** 表的叶子节点下。可保存的统计值个数由 **buckets number** 参数决定，当历史表的容量达到最大值时，系统会删除最早的记录来保存新的统计值。统计信息包括端口一个周期内收到的报文总数、广播报文总数和组播报文总数等。

在添加控制历史表项的过程中，如果指定的历史表容量超出了设备实际支持的历史表容量时，新的历史表项会被添加，但该表项对应生效的历史表容量为设备实际支持的历史表容量，可以使用 **display rmon history** 命令来查看配置结果。

如果在创建历史控制表项时，指定的采样间隔（**interval sampling-interval**）参数的值和该接口下已经存在的历史控制表项对应的该项参数值相同时，系统将认为这两个表项的配置相同，创建操作失败。

同一接口下可以创建多个历史控制表项。

## 【举例】

# 创建索引号为 1，表容量为 10，采样时间为 5 秒的历史控制表项，创建者为 user1。

```
<Sysname> system-view
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] rmon history 1 buckets 10 interval 5 owner user1
```

## 【相关命令】

- **display rmon history**

### 1.1.10 rmon prialarm

**rmon prialarm** 命令用来创建扩展告警表项。

**undo rmon prialarm** 命令用来删除指定的扩展告警表项。

## 【命令】

```
rmon prialarm entry-number prialarm-formula prialarm-des sampling-interval
{ absolute | delta } [ startup-alarm { falling | rising | rising-falling } ]
rising-threshold threshold-value1 event-entry1 falling-threshold
threshold-value2 event-entry2 entrytype { forever | cycle cycle-period }
[ owner text ]

undo rmon prialarm entry-number
```

## 【缺省情况】

扩展告警表中不存在表项。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**entry-number**: 扩展告警表项的索引号，取值范围为 1~65535。

**prialarm-formula**: 对告警变量进行计算的告警公式，为 1~255 个字符的字符串。公式中的告警变量必须以 OID 来表示，OID 表达式必须以小数点开始，例如(.1.3.6.1.2.1.2.1.10.1)\*8；运算公式由用户定义，可以使用加减乘除四种运算方法对告警变量进行运算，该运算公式的结果取值为长整型数，但不支持负数的输入。用户在编写公式的时候需要注意，公式中每一步的运算结果都不能超过长整型的表达范围，否则可能会得出错误的结果。

**prialarm-des**: 对该告警的描述，为 1~127 个字符的字符串，区分大小写。

**sampling-interval**: 采样间隔时间，取值范围为 10~65535，单位为秒。

**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。

**startup-alarm**: 表示初次采样时，如果达到或超出阈值，触发的告警类型。如果未指定本参数，触发 rising-falling 类型告警。

**rising**: 表示只触发上限告警。

**falling**: 表示只触发下限告警。

**rising-falling**: 表示触发上限或下限告警。

**rising-threshold threshold-value1 event-entry1**: 设置超上限参数，**threshold-value1** 表示上限阈值，取值范围为-2147483648~2147483647；**event-entry1** 表示上限阈值相应的事件索引号，取值范围为 0~65535（0 表示没有对应的事件，告警被触发后不会采取任何事件动作）。

**falling-threshold threshold-value2 event-entry2**: 设置下限参数，**threshold-value2** 表示下限阈值，取值范围为-2147483648~2147483647；**event-entry2** 表示下限阈值相应的事件索引号，取值范围为 0~65535（0 表示没有对应的事件，告警被触发后不会采取任何事件动作）。

**forever**: 本告警实例存活类型为永久。

**cycle cycle-period**: 本告警实例的存活时间，单位为秒，取值范围 0~4294967。

**owner text**: 该表项的创建者，为 1~127 个字符的字符串，区分大小写。

### 【使用指导】

用户定义了扩展告警表项后，系统先对定义的扩展告警公式中的告警变量按照定义的时间间隔进行采样，再将采样值按照定义的运算公式进行计算，最后将计算结果和设定的阈值进行比较，并执行相应的处理过程。

在添加扩展告警表项之前，需要通过 **rmon event** 命令定义好扩展告警表项中引用的事件。

如果在新建表项时，指定的告警变量公式（*prialarm-formula*）、采样间隔（*sampling-interval*）、采样类型（**absolute** 或 **delta**）、上限阈值（*threshold-value1*）和下限阈值（*threshold-value2*）五项参数的值和已经存在的扩展告警表项对应的五项参数值完全相同时，系统将认为这两个表项的配置相同，创建操作失败。

用户最多可以定义 50 个扩展告警表项。

### 【举例】

# 使用扩展告警对接口接收到的广播报文比率进行监控。

在扩展告警表中添加索引号为 1 的表项，对相应告警变量以公式 (.1.3.6.1.2.1.16.1.1.1.6.1\*100/.1.3.6.1.2.1.16.1.1.1.5.1) 运算，对该公式中涉及的变量以 10 秒的采样间隔进行绝对值采样。上限告警值为 80 对应事件 1（将事件记录在日志表中），下限告警值为 5 对应事件 2（不需要采取措施），表项的存活时间为永远（**forever**），创建者为 *user1*。（广播报文比率的计算公式为：接口接收到的广播报文总数/接口接收到的总报文数，该公式由用户自行定义）

```
<Sysname> system-view
[Sysname] rmon event 1 log
[Sysname] rmon event 2 none
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] rmon statistics 1
[Sysname-GigabitEthernet1/0/1] quit
[Sysname] rmon prialarm 1 (.1.3.6.1.2.1.16.1.1.1.6.1*100/.1.3.6.1.2.1.16.1.1.1.5.1)
BroadcastPktsRatioOfGE1/0/1 10 absolute rising-threshold 80 1 falling-threshold 5 2
entrytype forever owner user1
```

1.3.6.1.2.1.16.1.1.1.6.1 是节点 **etherStatsBroadcastPkts.1** 的 OID，1.3.6.1.2.1.16.1.1.1.5.1 是节点 **etherStatsPkts.1** 的 OID。.1 与接口统计表项的编号一致，如果创建的是“**rmon statistics 5**”，则对应需要使用 1.3.6.1.2.1.16.1.1.1.6.5 和 1.3.6.1.2.1.16.1.1.1.5.5。

### 【相关命令】

- **display rmon prialarm**
- **rmon event**

#### 1.1.11 rmon statistics

**rmon statistics** 命令用来创建统计表项。

**undo rmon statistics** 命令用来删除指定的统计表项。

### 【命令】

```
rmon statistics entry-number [ owner text ]
undo rmon statistics entry-number
```

### 【缺省情况】

统计表中不存在表项。

### 【视图】

以太网接口视图

### 【缺省用户角色】

network-admin

### 【参数】

**entry-number**: 统计表项的索引号，取值范围为 1～65535。

**owner text**: 该表项的创建者，为 1～127 个字符的字符串，区分大小写。

### 【使用指导】

当需要统计某个以太网端口的累加数据时，需要建立统计表。统计信息包括网络冲突数、CRC 校验错误报文数、过小（或超大）的数据报文数、广播、多播的报文数以及接收字节数、接收报文数等。设备重启时，会清除该统计信息。

用户可以通过 **display rmon statistics** 命令来显示统计表项的信息。

每个接口下只能定义一个统计表项。

### 【举例】

# 在统计表中添加 GigabitEthernet1/0/1 的统计表项，表项的索引号为 20，创建者为 user1。

```
<Sysname> system-view
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] rmon statistics 20 owner user1
```

### 【相关命令】

- **display rmon statistics**

# 目 录

1 NETCONF .....	1-1
1.1 NETCONF配置命令 .....	1-1
1.1.1 netconf capability specific-namespace.....	1-1
1.1.2 netconf idle-timeout.....	1-1
1.1.3 netconf log .....	1-2
1.1.4 netconf soap acl .....	1-4
1.1.5 netconf soap domain .....	1-5
1.1.6 netconf soap dscp.....	1-6
1.1.7 netconf soap enable.....	1-7
1.1.8 netconf ssh server enable.....	1-7
1.1.9 netconf ssh server port .....	1-8
1.1.10 xml .....	1-9

# 1 NETCONF

## 1.1 NETCONF配置命令



说明

设备运行于 FIPS 模式时，本特性部分配置相对于非 FIPS 模式有所变化，具体差异请见本文相关描述。有关 FIPS 模式的详细介绍请参见“安全配置指导”中的“FIPS”。

### 1.1.1 netconf capability specific-namespace

**netconf capability specific-namespace** 命令用来配置设备使用专用命名空间。

**undo netconf capability specific-namespace** 命令用来恢复缺省情况。

#### 【命令】

```
netconf capability specific-namespace
undo netconf capability specific-namespace
```

#### 【缺省情况】

NETCONF 使用共用命名空间。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【使用指导】

设备支持共用命名空间和专用命名空间，两种类型的命名空间互不兼容。客户端与设备必须使用相同的命名空间才能建立会话。缺省情况下，设备使用共用命名空间。如果客户端不支持共用命名空间，则需要配置本命令，让设备和客户端建立会话时使用专用命名空间。

配置该命令后，用户需要重新建立 NETCONF 会话，以使客户端和设备使用相同的命名空间。

#### 【举例】

# 配置设备使用专用命名空间。

```
<Sysname> system-view
[Sysname] netconf capability specific-namespace
```

### 1.1.2 netconf idle-timeout

**netconf idle-timeout** 命令用来配置 NETCONF 会话超时时间。

**undo netconf idle-timeout** 命令用来恢复缺省情况。

## 【命令】

```
netconf { soap | agent } idle-timeout minute  
undo netconf { soap | agent } idle-timeout
```

## 【缺省情况】

基于 NETCONF over SOAP over HTTP 和 NETCONF over SOAP over HTTPS 的会话超时时间为 10 分钟。

基于 NETCONF over SSH、NETCONF over Telnet 和 NETCONF over Console 的会话超时时间为 0，即不超时。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**soap**: 用来配置基于 NETCONF over SOAP over HTTP 和 NETCONF over SOAP over HTTPS 的会话超时时间。

**agent**: 用来配置基于 NETCONF over SSH、NETCONF over Telnet 和 NETCONF over Console 的会话超时时间。

**minute**: 表示空闲超时时间，单位为分钟，取值范围为：

- 对于基于 NETCONF over SOAP over HTTP 和 NETCONF over SOAP over HTTPS 的会话，取值范围为 1～999。
- 对于基于 NETCONF over SSH、NETCONF over Telnet 和 NETCONF over Console 的会话，取值范围为 0～999，0 表示不超时。

## 【使用指导】

用户通过 Telnet、SSH、Console、或配置工具与设备建立 NETCONF 连接，如果在超时时间内，用户与设备无任何 NETCONF 报文交互，达到超时时间后，设备将断开与用户配置端的 NETCONF 连接。

## 【举例】

# 配置基于 NETCONF over SOAP over HTTP 和 NETCONF over SOAP over HTTPS 的会话超时时间为 20 分钟。

```
<Sysname> system-view  
[Sysname] netconf soap idle-timeout 20
```

### 1.1.3 netconf log

**netconf log** 命令用来配置 NETCONF 日志功能。

**undo netconf log** 命令用来取消指定类型的 NETCONF 日志配置。



## 【命令】

```
netconf log source { all | { agent | soap | web } * } { protocol-operation  
{ all | { action | config | get | session | set | syntax | others } * } |  
row-operation | verbose }  
undo netconf log source { all | { agent | soap | web } * } { protocol-operation  
{ all | { action | config | get | session | set | syntax | others } * } |  
row-operation | verbose }
```

## 【缺省情况】

未配置 NETCONF 日志功能。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**source:** 表示需要输出日志的 NETCONF 操作来源:

- **all:** 表示输出所有来源的 NETCONF 日志。
- **agent:** 表示输出源自 Telnet/SSH/NETCONF over SSH/Console 类型的 NETCONF 日志。
- **soap:** 表示输出源自 SOAP over HTTP 和 SOAP over HTTPS 类型的 NETCONF 日志。
- **web:** 表示输出源自本设备 Web 类型的 NETCONF 日志。

**protocol-operation:** 表示按 NETCONF 操作类型输出日志:

- **all:** 表示所有 NETCONF 操作类型。
- **action:** 表示 action 操作。
- **config:** 表示配置类的 NETCONF 操作，包括 CLI、save、load、rollback、save-point、lock、unlock。
- **get:** 表示获取数据的 NETCONF 操作，包括 get、get-bulk、get-config、get-bulk-config、get-sessions。
- **session:** 表示与会话相关的 NETCONF 操作，包括 kill-session、close-session、hello。
- **set:** 表示所有 edit-config 操作。
- **syntax:** 表示包含 xml 和 schema 的格式错误的请求。
- **others:** 表示除 action、config、get、set、session、set、syntax 外，其他类型的 NETCONF 操作。

**row-operation:** 表示按 NETCONF 行操作输出日志。该参数仅支持 action 和 set 操作。

**verbose:** 表示输出 NETCONF 操作的详细日志信息。该参数仅对按操作类型输出日志方式生效:

- 对于请求类型的日志: 当请求报文格式正确时，设备在简要日志后输出源报文的内容。
- 对于处理结果类型的日志: 仅支持输出 **set** 的详细日志信息。即，当 edit-config 操作执行错误时输出详细错误信息。

## 【使用指导】

NETCONF 日志功能支持按 NETCONF 操作类型（**protocol-operation**）输出日志和按 NETCONF 行操作（**row-operation**）输出日志：

- 配置按 NETCONF 操作类型输出日志时，用户每下发一次 NETCONF 操作，设备输出该 NETCONF 操作、以及操作结果的日志。例如用户下发 NETCONF 操作，创建 ID 为 3~5 的 VLAN 时，设备输出的日志如下：

```
%Mar 21 17:11:34:479 2017 H3C XMLSOAP/6/XML_REQUEST: test from 192.168.100.198, session id 2,message-id 100, receive edit-config request.
```

```
%Mar 21 17:11:34:483 2017 H3C XMLSOAP/6/EDIT-CONFIG: test from 192.168.100.198, session id 2,message-id 100, execute success.
```

- 配置按 NETCONF 行操作输出日志时，用户下发一次 NETCONF 操作后，设备输出该操作中每个请求行操作的日志。例如用户下发 NETCONF 操作，创建 ID 为 3~5 的 VLAN 时，设备输出的日志如下：

```
%Mar 31 17:50:02:608 2017 H3C XMLSOAP/6/EDIT-CONFIG: User (test, 192.168.100.20, session ID 1), message ID=100, operation=create VLAN/VLANs (ID=3), result=Succeeded. No attributes.
```

```
%Mar 31 17:50:02:609 2017 H3C XMLSOAP/6/EDIT-CONFIG: User (test, 192.168.100.20, session ID 1), message ID=100, operation=create VLAN/VLANs (ID=4), result=Succeeded. No attributes.
```

```
%Mar 31 17:50:02:611 2017 H3C XMLSOAP/6/EDIT-CONFIG: User (test, 192.168.100.20, session ID 1), message ID=100, operation=create VLAN/VLANs (ID=5), result=Succeeded. No attributes.
```

设备先将 NETCONF 日志发送到信息中心，根据信息中心的输出规则将日志信息输出到不同方向。信息中心的相关配置请参见“网络管理和监控配置指导”中的“信息中心”。

## 【举例】

# 配置设备输出源自 agent 的 edit-config 类型的操作日志。

```
<Sysname> system-view
[sysname] netconf log source agent protocol-operation set
```

### 1.1.4 netconf soap acl

**netconf soap acl** 命令用来配置 NETCONF over SOAP 关联 IPv4 ACL。

**undo netconf soap acl** 命令用来恢复缺省情况。

## 【命令】

非 FIPS 模式下：

```
netconf soap { http | https } acl { ipv4-acl-number | name ipv4-acl-name }
undo netconf soap { http | https } acl
```

FIPS 模式下：

```
netconf soap https acl { ipv4-acl-number | name ipv4-acl-name }
undo netconf soap https acl
```

## 【缺省情况】

未配置 NETCONF over SOAP 关联 IPv4 ACL。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**ipv4-acl-number**: IPv4 ACL 的编号，取值范围为 2000~2999。

**http**: 开启基于 HTTP 的 NETCONF over SOAP 关联 IPv4 ACL。

**https**: 开启基于 HTTPS 的 NETCONF over SOAP 关联 IPv4 ACL。

**name ipv4-acl-name**: 指定 IPv4 ACL 的名称。*ipv4-acl-name* 表示 IPv4 ACL 的名称，为 1~63 个字符的字符串，不区分大小写，必须以英文字母 a~z 或 A~Z 开头。为避免混淆，IPv4 ACL 的名称不允许使用英文单词 **all**。仅当指定名称的 IPv4 ACL 存在且为基本 IPv4 ACL 时生效。

## 【使用指导】

配置 SOAP 服务与 IPv4 ACL 关联后，只有 IPv4 ACL 允许通过的 NETCONF 客户端能够通过 SOAP 方式登录设备。不匹配 IPv4 ACL 或 IPv4 ACL 拒绝通过的 NETCONF 客户端将不能通过 SOAP 方式登录设备。

多次执行 **netconf soap http acl** 命令，最后一次执行的命令生效。**netconf soap https acl** 命令亦然。

## 【举例】

# 配置基于 HTTP 的 NETCONF over SOAP 功能与 IPv4 ACL 2001 关联，只允许 10.10.0.0/16 网段的客户端访问设备。

```
<Sysname> system-view
[Sysname] acl basic 2001
[Sysname-acl-ipv4-basic-2001] rule permit source 10.10.0.0 0.0.255.255
[Sysname-acl-ipv4-basic-2001] quit
[Sysname] netconf soap http acl 2001
```

### 1.1.5 netconf soap domain

**netconf soap domain** 命令用来配置 NETCONF 用户的强制认证域。

**undo netconf soap domain** 命令用来恢复缺省情况。

## 【命令】

**netconf soap domain** *domain-name*

**undo netconf soap domain** *domain-name*

## 【缺省情况】

未配置 NETCONF 用户的强制认证域。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

### 【参数】

**domain-name:** ISP 域名，为 1~255 个字符的字符串，不区分大小写。关于 ISP 域的详细介绍请参见“安全配置指导”中的“AAA”。

### 【使用指导】

多次执行本命令，最后一次执行的命令生效。

NETCONF 支持使用如下方式指定认证域：

- 通过 **netconf soap domain** 命令配置强制认证域。配置该命令后，所有用户都会使用强制认证域进行认证。
- 在 SOAP 请求的<UserName>中携带认证域信息。SOAP 请求中携带的认证域仅对当前请求生效。通过 **netconf soap domain** 命令配置强制认证域后，SOAP 请求中携带的认证域不生效。

### 【举例】

# 配置 NETCONF 用户的强制认证域为 my-domain。

```
<Sysname> system-view
[Sysname] netconf soap domain my-domain
```

## 1.1.6 netconf soap dscp

**netconf soap dscp** 命令用来配置设备发送的 SOAP 报文的 DSCP 优先级。

**undo netconf soap dscp** 命令用来恢复缺省情况。

### 【命令】

非 FIPS 模式下：

```
netconf soap { http | https } dscp dscp-value
undo netconf soap { http | https } dscp
```

FIPS 模式下：

```
netconf soap https dscp dscp-value
undo netconf soap https dscp
```

### 【缺省情况】

设备发送的 SOAP 报文的 DSCP 优先级为 0。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**dscp-value:** 指定设备发送的基于 HTTP 的 SOAP 报文的 DSCP 优先级，取值范围为 0~63。

**http:** 设备发送基于 HTTP 的 SOAP 报文的 DSCP 优先级。

**https:** 设备发送基于 HTTPS 的 SOAP 报文的 DSCP 优先级。

### 【使用指导】

DSCP 优先级封装在 IP 报文中的 ToS 字段，用来表示报文自身的优先级，设备根据该优先级决定报文传输的优先程度。

### 【举例】

# 配置设备发送的基于 HTTP 的 SOAP 报文的 DSCP 优先级为 30。

```
<Sysname> system-view
[Sysname] netconf soap http dscp 30
```

## 1.1.7 netconf soap enable

**netconf soap enable** 命令用来开启 NETCONF over SOAP 功能。

**undo netconf soap enable** 命令用来关闭 NETCONF over SOAP 功能。

### 【命令】

非 FIPS 模式下：

```
netconf soap { http | https } enable
undo netconf soap { http | https } enable
```

FIPS 模式下：

```
netconf soap https enable
undo netconf soap https enable
```

### 【缺省情况】

NETCONF over SOAP 处于关闭状态。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**http**：开启基于 HTTP 的 NETCONF over SOAP 功能。

**https**：开启基于 HTTPS 的 NETCONF over SOAP 功能。

### 【使用指导】

配置该命令后，表示设备能够解析这样的 HTTP/HTTPS 报文，报文中的数据为 SOAP 封装过的 NETCONF 指令。

### 【举例】

# 开启基于 HTTP 的 SOAP 功能。

```
<Sysname> system-view
[Sysname] netconf soap http enable
```

## 1.1.8 netconf ssh server enable

**netconf ssh server enable** 命令用来开启 NETCONF over SSH 的接入方式。

**undo netconf ssh server enable** 命令用来关闭 NETCONF over SSH 的接入方式。

【命令】

```
netconf ssh server enable
undo netconf ssh server enable
```

【缺省情况】

NETCONF over SSH 的接入方式处于关闭状态。

【视图】

系统视图

【缺省用户角色】

network-admin

【使用指导】

用户配置该命令后，可以利用 SSH 客户端通过 SSH 子系统的方式接入设备的 NETCONF 系统，然后直接进入 NETCONF 配置模式，而不用手工输入 XML 命令。

配置该命令前必须在设备上把 SSH 连接终端的认证方式设置为 **scheme**，支持 NETCONF over SSH 的客户端才能连接到 NETCONF 系统，目前只支持用 **urn:ietf:params:netconf:base:1.0**（设备与终端共同支持的能力集）连接系统。

【举例】

# 开启 NETCONF over SSH 的接入方式。

```
<Sysname> system-view
[Sysname] netconf ssh server enable
```

### 1.1.9 netconf ssh server port

**netconf ssh server port** 命令用来设置 NETCONF over SSH 接入方式的监听端口号。

**undo netconf ssh server port** 命令用来恢复缺省情况。

【命令】

```
netconf ssh server port port-number
undo netconf ssh server port
```

【缺省情况】

NETCONF over SSH 接入方式的监听端口为 830。

【视图】

系统视图

【缺省用户角色】

network-admin

【参数】

*port-number*：基于 NETCONF over SSH 的接入方式的监听端口，取值范围为 1～65535。

### 【使用指导】

用户可以在必要时使用此命令来重新配置一个端口作为 NETCONF 子系统的监听端口,但由于 SSH 服务使用共享端口的方式来分配监听端口,为了保证正常使用,必须保证分配的端口不和其他使用的端口冲突。

### 【举例】

# 配置 NETCONF over SSH 接入方式的监听端口为 800。

```
<sysname> system-view
[sysname] netconf ssh server port 800
```

## 1.1.10 xml

**xml** 命令用来进入 XML 视图。

### 【命令】

**xml**

### 【视图】

用户视图

### 【缺省用户角色】

network-admin  
network-operator

### 【使用指导】

进入 XML 视图后可以输入 NETCONF 指令来配置或者获取系统数据。用户登录时使用的角色不同,可执行的 NETCONF 操作也不同:

- network-admin 可执行全部操作。
- network-operator 可执行 get、get-bulk、get-config、get-bulk-config、get-sessions、close-session 操作。

进入 XML 视图后,用户需要严格按照 NETCONF 报文格式将报文拷贝、粘贴到 XML 视图中。在 NETCONF 配置过程中,请勿进行其他任何操作(例如手工输入 NETCONF 报文),否则可能导致 NETCONF 配置失败。

在 XML 视图下进行 NETCONF 配置时,NETCONF 报文最后需要添加“]]>]]>”,否则设备无法识别。

进入 XML 视图后,设备将自动向客户端发送自身支持的 NETCONF 能力集,此时,客户端需要向设备发送自身支持的 NETCONF 能力集,完成能力交换后,用户即可使用客户端对设备进行配置。用户输入的 NETCONF 指令必须符合 XML 语言格式要求和《NETCONF XML API 手册》中的语法、语义要求。建议使用第三方软件来协助生成 NETCONF 指令,命令行手工输入方式通常用于研发和测试环境。

退出 XML 视图时需要使用相关的 NETCONF 指令,不能使用 **quit**。

在 XML 模式下终止任务的快捷键有重置缓存的功能,快捷键之前的内容都会被清除掉。如果在用户线/用户线类视图下使用 **escape-key** 命令配置了终止任务的快捷键(缺省为 Ctrl+C),可能会影响 XML 视图下相关配置。例如:在用户线视图下配置了 **escape-key a**,当 NETCONF 指令中

含有字符 ‘a’ 时，其实只有 NETCONF 指令最后一个 ‘a’ 之后的内容能够得到处理；当 NETCONF 指令中不含有字符 ‘a’ 时，则对 XML 视图下的配置没有影响。

### 【举例】

# 进入 XML 视图。

```
<Sysname> xml
<?xml version="1.0" encoding="UTF-8"?><hello
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"><capabilities><capability>urn:ietf:params:netconf:base:1.1</capability><capability>urn:ietf:params:netconf:writable-running</capability><capability>urn:ietf:params:netconf:capability:notification:1.0</capability><capability>urn:ietf:params:netconf:capability:validate:1.1</capability><capability>urn:ietf:params:netconf:capability:interleave:1.0</capability><capability>urn:h3c:params:netconf:capability:h3c-netconf-ext:1.0</capability></capabilities><session-id>1</session-id></hello>]]>]]>
```

# 进行能力交换。

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

# 退出 XML 视图。

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <close-session>
</close-session>
</rpc>]]>]]>
<Sysname>
```



# 目 录

1 CWMP .....	1-1
1.1 CWMP配置命令 .....	1-1
1.1.1 cwmp .....	1-1
1.1.2 cwmp acs default password .....	1-1
1.1.3 cwmp acs default url .....	1-2
1.1.4 cwmp acs default username .....	1-3
1.1.5 cwmp acs password .....	1-4
1.1.6 cwmp acs url .....	1-4
1.1.7 cwmp acs username .....	1-5
1.1.8 cwmp cpe connect interface .....	1-6
1.1.9 cwmp cpe connect retry .....	1-7
1.1.10 cwmp cpe inform interval .....	1-7
1.1.11 cwmp cpe inform interval enable .....	1-8
1.1.12 cwmp cpe inform time .....	1-9
1.1.13 cwmp cpe password .....	1-9
1.1.14 cwmp cpe provision-code .....	1-10
1.1.15 cwmp cpe stun enable .....	1-11
1.1.16 cwmp cpe username .....	1-12
1.1.17 cwmp cpe wait timeout .....	1-12
1.1.18 cwmp enable .....	1-13
1.1.19 display cwmp configuration .....	1-14
1.1.20 display cwmp status .....	1-15
1.1.21 ssl client-policy .....	1-16

# 1 CWMP

## 1.1 CWMP配置命令

### 1.1.1 cwmp

**cwmp** 命令用来进入 CWMP 视图。

#### 【命令】

**cwmp**

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【举例】

```
# 进入 CWMP 视图。
<Sysname> system-view
[Sysname] cwmp
```

#### 【相关命令】

- **cwmp enable**

### 1.1.2 cwmp acs default password

**cwmp acs default password** 命令用来配置 CPE 连接 ACS 的缺省密码。

**undo cwmp acs default password** 命令用来恢复缺省情况。

#### 【命令】

```
cwmp acs default password { cipher | simple } string
undo cwmp acs default password
```

#### 【缺省情况】

未配置 CPE 连接 ACS 的缺省密码。

#### 【视图】

CWMP 视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**cipher:** 以密文方式设置密码。

**simple:** 以明文方式设置密码，该密码将以密文形式存储。

*string*: 密码字符串，区分大小写。明文密码为 1~255 个字符的字符串；密文密码为 33~373 个字符的字符串。

#### 【使用指导】

一个 CPE 只能配置一个连接 ACS 的密码和缺省密码。

该配置为可选配置，可以只用用户名验证，但 ACS 和 CPE 上的配置必须一致。

多次执行本命令，最后一次执行的命令生效。

#### 【举例】

# 配置 CPE 连接 ACS 的缺省密码为 newpsw。

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

```
[Sysname] cwmp
```

```
[Sysname-cwmp] cwmp acs default password simple newpsw
```

#### 【相关命令】

- **cwmp acs default url**
- **cwmp acs default username**

### 1.1.3 cwmp acs default url

**cwmp acs default url** 命令用来配置 CPE 连接 ACS 的缺省 URL。

**undo cwmp acs default url** 命令用来恢复缺省情况。

#### 【命令】

```
cwmp acs default url url
```

```
undo cwmp acs default url
```

#### 【缺省情况】

未配置 CPE 连接 ACS 的缺省 URL。

#### 【视图】

CWMP 视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*url*: CPE 连接 ACS 的缺省 URL，为 8~255 个字符的字符串，格式必须为：  
`http://host[:port]/path` 或者 `https://host[:port]/path`。

#### 【使用指导】

当用户没有为 ACS 配置 URL 地址，也没有通过 DHCP 服务器获取到 ACS 的 URL 地址时，设备会尝试和 ACS 的缺省 URL 建立 CWMP 连接。

一个 CPE 只能配置一个连接 ACS 的 URL 和缺省 URL。

多次执行本命令，最后一次执行的命令生效。

#### 【举例】

# 配置 CPE 连接 ACS 的缺省 URL 为 `http://www.acs.com:9090`。

```
<Sysname> system-view
[Sysname] cwmp
[Sysname-cwmp] cwmp acs default url http://www.acs.com:9090
```

#### 【相关命令】

- **cwmp acs default password**
- **cwmp acs default username**

### 1.1.4 cwmp acs default username

**cwmp acs default username** 命令用来配置 CPE 连接 ACS 的缺省用户名。

**undo cwmp acs default username** 命令用来恢复缺省情况。

#### 【命令】

```
cwmp acs default username username
undo cwmp acs default username
```

#### 【缺省情况】

未配置 CPE 连接 ACS 的缺省用户名。

#### 【视图】

CWMP 视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**username**: CPE 向 ACS 的缺省 URL 发送连接请求时携带的用户名, 为 1~255 个字符的字符串, 区分大小写。

#### 【使用指导】

CWMP 建立连接时, 使用的用户名和密码必须和 ACS 上创建的用户名和密码一致, 否则, 连接建立失败。

一个 CPE 只能配置一个连接 ACS 的用户名和缺省用户名。

多次执行本命令, 最后一次执行的命令生效。

#### 【举例】

# 配置 CPE 连接 ACS 的缺省用户名为 newname。

```
<Sysname> system-view
[Sysname] cwmp
[Sysname-cwmp] cwmp acs default username newname
```

#### 【相关命令】

- **cwmp acs default password**
- **cwmp acs default url**

### 1.1.5 cwmp acs password

**cwmp acs password** 命令用来配置 CPE 连接 ACS 的密码。

**undo cwmp acs password** 命令用来恢复缺省情况。

#### 【命令】

```
cwmp acs password { cipher | simple } string
undo cwmp acs password
```

#### 【缺省情况】

未配置 CPE 连接 ACS 的密码。

#### 【视图】

CWMP 视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**cipher**: 以密文方式设置密码。

**simple**: 以明文方式设置密码，该密码将以密文形式存储。

**string**: 密码字符串，区分大小写。明文密码为 1~255 个字符的字符串，密文密码为 33~373 个字符的字符串。

#### 【使用指导】

该配置为可选配置，可以只用用户名验证，但 ACS 和 CPE 上的配置必须一致。

一个 CPE 只能配置一个连接 ACS 的密码和缺省密码。

多次执行本命令，最后一次执行的命令生效。

#### 【举例】

```
# 配置 CPE 连接 ACS 的密码为 newpsw。
<Sysname> system-view
[Sysname] cwmp
[Sysname-cwmp] cwmp acs password simple newpsw
```

#### 【相关命令】

- **cwmp acs url**
- **cwmp acs username**

### 1.1.6 cwmp acs url

**cwmp acs url** 命令用来配置 CPE 连接 ACS 的 URL。

**undo cwmp acs url** 命令用来恢复缺省情况。

#### 【命令】

```
cwmp acs url url
undo cwmp acs url
```

### 【缺省情况】

未配置 CPE 连接 ACS 的 URL。

### 【视图】

CWMP 视图

### 【缺省用户角色】

network-admin

### 【参数】

**url**：指定 CPE 连接 ACS 的 URL，为 8～255 个字符的字符串，格式必须为：  
`http://host[:port]/path` 或者 `https://host[:port]/path`。

### 【使用指导】

配置该命令后，如果有连接需求，则设备会向该命令指定的 ACS 发起 CWMP 连接请求。

ACS 有三种指定方式，按照优先级从高到底依次为：通过该命令指定，通过 DHCP 协议从 DHCP 服务器获取，通过 `cwmp acs default url` 命令指定。当通过优先级高的方式获取不到 URL 时，再尝试优先级低的方式。

一个 CPE 只能配置一个连接 ACS 的 URL 和缺省 URL。

多次执行本命令，最后一次执行的命令生效。

### 【举例】

# 配置 CPE 连接 ACS 的 URL 为 `http://www.acs.com:9090`。

```
<Sysname> system-view
[Sysname] cwmp
[Sysname-cwmp] cwmp acs url http://www.acs.com:9090
```

## 1.1.7 cwmp acs username

**cwmp acs username** 命令用来配置 CPE 连接 ACS 的用户名。

**undo cwmp acs username** 命令用来恢复缺省情况。

### 【命令】

```
cwmp acs username username
undo cwmp acs username
```

### 【缺省情况】

未配置 CPE 连接 ACS 的用户名。

### 【视图】

CWMP 视图

### 【缺省用户角色】

network-admin

### 【参数】

**username**：CPE 向 ACS 的 URL 发送连接请求时携带的用户名，为 1～255 个字符的字符串，区分大小写。

### 【使用指导】

一个 CPE 只能配置一个连接 ACS 的用户名和缺省用户名。  
多次执行本命令，最后一次执行的命令生效。

### 【举例】

```
# 配置 CPE 连接 ACS 的用户名为 newname。
<Sysname> system-view
[Sysname] cwmp
[Sysname-cwmp] cwmp acs username newname
```

### 【相关命令】

- **cwmp acs password**

## 1.1.8 cwmp cpe connect interface

**cwmp cpe connect interface** 命令用来设置 CPE 上用于连接 ACS 的接口。  
**undo cwmp cpe connect interface** 命令用来恢复缺省情况。

### 【命令】

```
cwmp cpe connect interface interface-type interface-number
undo cwmp cpe connect interface
```

### 【缺省情况】

### 【视图】

CWMP 视图

### 【缺省用户角色】

network-admin

### 【参数】

*interface-type interface-number*: 指定 CPE 上用于连接 ACS 的接口类型和编号。

### 【使用指导】

CWMP 连接接口指的是 CPE 上用于连接 ACS 的接口。CPE 会在 Inform 报文中携带 CWMP 连接接口的 IP 地址,要求 ACS 通过此 IP 地址和自己建立连接;相应的,ACS 会向该 IP 地址回复 Inform 响应报文。

通常情况下,系统会采用一定的机制去自动获取一个 CWMP 连接接口,但如果获取的 CWMP 连接接口不是 CPE 和 ACS 实际相连的接口时(比如 CPE 存在多个接口,且自动获取的接口上的 IP 地址与 ACS 的 IP 地址不在同一个网段时,表示当前获取的 CPE 接口和 ACS 接口不是实际相连),就会导致 CWMP 连接建立失败。因此,在这种情况下需要手工指定 CWMP 连接接口。

### 【举例】

```
# 配置 CPE 上与 ACS 连接的接口为 GigabitEthernet1/0/1。
<Sysname> system-view
[Sysname] cwmp
[Sysname-cwmp] cwmp cpe connect interface gigabitethernet 1/0/1
```

### 1.1.9 cwmp cpe connect retry

**cwmp cpe connect retry** 命令用来配置建立 CWMP 连接时，连接失败后自动重新连接的次数。

**undo cwmp cpe connect retry** 命令用来恢复缺省情况。

#### 【命令】

```
cwmp cpe connect retry retries
```

```
undo cwmp cpe connect retry
```

#### 【缺省情况】

重新连接次数为无限次，即设备会一直按照一定周期向 ACS 发送连接请求。

#### 【视图】

CWMP 视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**retries**: 连接失败后自动重新连接的次数，取值范围为 0~100，0 表示不重新发起连接。

#### 【使用指导】

当 CPE 向 ACS 请求建立连接失败，或者在会话过程中连接异常中止（CPE 没有收到表示会话正常结束的报文）时，设备可以自动重新发起连接。

#### 【举例】

# 配置建立 CWMP 连接时，连接失败后自动重新连接为 5 次。

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

```
[Sysname] cwmp
```

```
[Sysname-cwmp] cwmp cpe connect retry 5
```

### 1.1.10 cwmp cpe inform interval

**cwmp cpe inform interval** 命令用来配置周期发送 Inform 报文的时间间隔。

**undo cwmp cpe inform interval** 命令用来恢复缺省情况。

#### 【命令】

```
cwmp cpe inform interval interval
```

```
undo cwmp cpe inform interval
```

#### 【缺省情况】

CPE 周期发送 Inform 报文的时间间隔为 600 秒。

#### 【视图】

CWMP 视图

#### 【缺省用户角色】

network-admin



### 【参数】

*interval*: 周期发送 Inform 报文的时间间隔, 取值范围为 10~86400, 单位为秒。

### 【使用指导】

CPE 与 ACS 之间连接的建立过程需要发送 Inform 报文。通过设置 Inform 报文发送参数, 可以触发 CPE 向 ACS 自动发起连接。

该命令用于设置 CPE 向 ACS 发送 Inform 报文的时间间隔。

只有在配置了 **cwmp cpe inform interval enable** 命令时, 该命令才会生效。

### 【举例】

# 配置 CPE 周期发送 Inform 报文的时间间隔为 3600 秒。

```
<Sysname> system-view
[Sysname] cwmp
[Sysname-cwmp] cwmp cpe inform interval enable
[Sysname-cwmp] cwmp cpe inform interval 3600
```

### 【相关命令】

- **cwmp cpe inform interval enable**

#### 1.1.11 cwmp cpe inform interval enable

**cwmp cpe inform interval enable** 命令用来开启 CPE 周期发送 Inform 报文功能。

**undo cwmp cpe inform interval enable** 命令用来关闭 CPE 周期发送 Inform 报文功能。

### 【命令】

```
cwmp cpe inform interval enable
undo cwmp cpe inform interval enable
```

### 【缺省情况】

CPE 周期发送 Inform 报文功能处于关闭状态。

### 【视图】

CWMP 视图

### 【缺省用户角色】

network-admin

### 【使用指导】

开启 CPE 周期发送 Inform 报文功能, 当设定的周期达到时, CPE 会自动发送 Inform 报文与 ACS 建立连接。

### 【举例】

# 开启 CPE 周期发送 Inform 报文功能。

```
<Sysname> system-view
[Sysname] cwmp
[Sysname-cwmp] cwmp cpe inform interval enable
```

### 【相关命令】

- `cwmp cpe inform interval`

### 1.1.12 cwmp cpe inform time

`cwmp cpe inform time` 命令用来配置 CPE 在指定时刻发送一次 Inform 报文。

`undo cwmp cpe inform time` 命令用来恢复缺省情况。

### 【命令】

`cwmp cpe inform time time`

`undo cwmp cpe inform time`

### 【缺省情况】

未配置 CPE 定时发送 Inform 报文的时间。

### 【视图】

CWMP 视图

### 【缺省用户角色】

network-admin

### 【参数】

*time*: 指定 CPE 发送一次 Inform 报文的日期和时间，格式为：*yyyy-mm-ddThh:mm:ss*，取值范围为 1970-01-01T00:00:00~2035-12-31T23:59:59，该时间必须大于系统当前时间。

### 【使用指导】

CPE 与 ACS 之间连接的建立过程需要发送 Inform 报文。通过设置 Inform 报文发送参数，可以触发 CPE 向 ACS 自动发起连接。

### 【举例】

# 配置 CPE 发送 Inform 报文的日期和时间 2012-12-01T20:00:00。

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

```
[Sysname] cwmp
```

```
[Sysname-cwmp] cwmp cpe inform time 2012-12-01T20:00:00
```

### 1.1.13 cwmp cpe password

`cwmp cpe password` 命令用来配置 ACS 连接 CPE 时的认证密码。

`undo cwmp cpe password` 命令用来恢复缺省情况。

### 【命令】

`cwmp cpe password { cipher | simple } string`

`undo cwmp cpe password`

### 【缺省情况】

未配置 ACS 连接 CPE 的密码。

## 【视图】

CWMP 视图

## 【缺省用户角色】

network-admin

## 【参数】

**cipher:** 以密文方式设置密码。

**simple:** 以明文方式设置密码，该密码将以密文形式存储。

**string:** 密码字符串，区分大小写。明文密码为 1~255 个字符的字符串，密文密码为 33~373 个字符的字符串。

## 【使用指导】

当 ACS 与 CPE 建立 CWMP 连接且通过用户名和密码进行认证时，ACS 会将用户名和密码发送给 CPE，以便设备对 ACS 的身份进行认证。设备根据本地配置的用户名和该密码验证 ACS 是否合法，如果验证成功，则建立连接，否则，不能建立连接。

该配置为可选配置，可以只用用户名验证，但 ACS 和 CPE 上的配置必须一致。

一个 ACS 只能配置一个连接 CPE 的认证密码。

多次执行本命令，最后一次执行的命令生效。

## 【举例】

```
# 配置 ACS 连接 CPE 密码为 newpsw。  
<Sysname> system-view  
[Sysname] cwmp  
[Sysname-cwmp] cwmp cpe password simple newpsw
```

## 【相关命令】

- **cwmp cpe username**

### 1.1.14 cwmp cpe provision-code

**cwmp cpe provision-code** 命令用来配置 CPE 的业务代码。

**undo cwmp cpe provision-code** 命令用来恢复缺省情况。

## 【命令】

```
cwmp cpe provision-code provision-code  
undo cwmp cpe provision-code
```

## 【缺省情况】

CPE 向 ACS 发送的 Inform 报文中携带的业务代码为“PROVISIONINGCODE”。

## 【视图】

CWMP 视图

## 【缺省用户角色】

network-admin

### 【参数】

*provision-code*: 设备向 ACS 发送 Inform 报文中携带的设备代码。为 1~64 个字符的字符串，必须为大写字母、数字或者“.”。

### 【使用指导】

当 CPE 与 ACS 之间建立连接时，CPE 需要在 Inform 报文中携带 *provision-code* 信息，ACS 根据此信息可以识别设备定制的业务以及相应的参数，以便更好地管理 CPE 设备。关于 ACS 对 *provision-code* 的支持情况，请参见 ACS 手册。

多次执行本命令，最后一次执行的命令生效。

### 【举例】

```
# 配置 CPE 的业务代码为 ABC20150714。
<Sysname> system
[Sysname] cwmp
[Sysname-cwmp] cwmp cpe provision-code ABC20150714
```

## 1.1.15 cwmp cpe stun enable

***cwmp cpe stun enable*** 命令用来开启 CPE 的 NAT 穿越功能。

***undo cwmp cpe stun enable*** 命令用来关闭 CPE 的 NAT 穿越功能。

### 【命令】

```
cwmp cpe stun enable
undo cwmp cpe stun enable
```

### 【缺省情况】

CPE 的 NAT 穿越功能处于关闭状态。

### 【视图】

CWMP 视图

### 【缺省用户角色】

network-admin

### 【使用指导】

无论 CPE 与 ACS 之间是否存在 NAT 网关，CPE 的主动连接请求都能到达 ACS。而当 CPE 与 ACS 之间存在 NAT 网关时，ACS 主动发起的连接请求不能到达 CPE。此时，可以在设备上开启 NAT 穿越功能，使得 ACS 的请求能够穿越网关。本特性的实现遵循 RFC 3489 定义的 STUN（Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)，NAT 的 UDP 简单穿越）。

CPE 在主动给 ACS 发连接请求的过程中，如果发现与 ACS 之间存在 NAT 网关，则会将获取到的经 NAT 绑定的公网的 IP 地址和端口号发送给 ACS。

为了保证 ACS 任意时刻主动发起的连接请求能够穿越 NAT 网关到达 CPE，CPE 必须维持 NAT 网关上的地址映射关系。

有关 NAT 的详细描述，请参见“三层技术-IP 业务配置指导”中的“NAT”。

### 【举例】

# 开启 CPE 的 NAT 穿越功能。

```
<Sysname> system-view
[Sysname] cwmp
[Sysname-cwmp] cwmp cpe stun enable
```

## 1.1.16 cwmp cpe username

**cwmp cpe username** 命令用来配置 ACS 连接 CPE 时的认证用户名。

**undo cwmp cpe username** 命令用来恢复缺省情况。

### 【命令】

```
cwmp cpe username username
undo cwmp cpe username
```

### 【缺省情况】

未配置 ACS 连接 CPE 的用户名。

### 【视图】

CWMP 视图

### 【缺省用户角色】

network-admin

### 【参数】

**username**: ACS 请求连接 CPE 时的认证用户名，为 1~255 个字符的字符串，区分大小写。

### 【使用指导】

当 ACS 向 CPE 发送连接请求且通过用户名和密码认证时，ACS 会将用户名和密码发送给设备，以便设备对 ACS 的身份进行认证。设备根据本地配置的用户名和该密码验证 ACS 是否合法，如果验证成功，则建立连接，否则，不能建立连接。

一个 ACS 只能配置一个连接 CPE 的认证用户名。

多次执行本命令，最后一次执行的命令生效。

### 【举例】

# 配置 ACS 连接 CPE 的用户名为 newname。

```
<Sysname> system-view
[Sysname] cwmp
[Sysname-cwmp] cwmp cpe username newname
```

### 【相关命令】

- **cwmp cpe password**

## 1.1.17 cwmp cpe wait timeout

**cwmp cpe wait timeout** 命令用来配置 CPE 无数据传输超时时间。

**undo cwmp cpe wait timeout** 命令用来恢复缺省情况。

#### 【命令】

```
cwmp cpe wait timeout seconds
undo cwmp cpe wait timeout
```

#### 【缺省情况】

无数据传输超时时间为 30 秒。

#### 【视图】

CWMP 视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*seconds*: 无数据传输超时时间，取值范围为 30~1800，单位为秒。

#### 【使用指导】

无数据传输超时时间主要用于以下两种情况：

- 在连接建立过程中，CPE 向 ACS 发送连接请求，但是经过无数据传输超时时间还没有收到响应报文，CPE 将认为连接失败。
- CWMP 连接建立后，如果 CPE 与 ACS 在无数据传输超时时间内一直没有报文的交互，CPE 将认为连接失效，并断开连接。

#### 【举例】

```
# 配置 CPE 无数据传输超时时间为 60 秒。
<Sysname> system-view
[Sysname] cwmp
[Sysname-cwmp] cwmp cpe wait timeout 60
```

### 1.1.18 cwmp enable

**cwmp enable** 命令用来开启 CWMP 功能。

**undo cwmp enable** 命令用来关闭 CWMP 功能。

#### 【命令】

```
cwmp enable
undo cwmp enable
```

#### 【缺省情况】

CWMP 功能处于关闭状态。

#### 【视图】

CWMP 视图

#### 【缺省用户角色】

network-admin

【使用指导】

开启 CWMP 后，CWMP 的其它配置才能生效。

【举例】

```
# 开启 CWMP 功能。
<Sysname> system-view
[Sysname] cwmp
[Sysname-cwmp] cwmp enable
```

【相关命令】

- **cwmp**

1.1.19 display cwmp configuration

**display cwmp configuration** 命令用来显示 CWMP 的配置信息。

【命令】

```
display cwmp configuration
```

【视图】

任意视图

【缺省用户角色】

network-admin  
network-operator

【举例】

```
# CWMP 开启，显示 CWMP 的配置信息。
<sysname> display cwmp configuration
CWMP state                : Enabled
ACS URL                   : http://www.acs.com:9090
ACS username              : newname
ACS default URL           : Null
ACS default username      : defname
Periodic inform           : Disabled
Inform interval           : 600s
Inform time               : None
Wait timeout              : 30s
Connection retries        : Unlimited
Source IP interface       : None
STUN state                : Disabled
SSL policy name           : Null
```

表1-1 display cwmp configuration 命令显示信息描述表

字段	描述
CWMP state	CWMP的开启状态: <ul style="list-style-type: none"><li>• Enabled: 表示已开启</li><li>• Disabled: 表示未开启</li></ul>

字段	描述
ACS default URL	CPE连接ACS的缺省URL，没有配置时显示为Null
ACS default username	CPE连接ACS的缺省用户名，没有配置时显示为Null
ACS URL	CPE连接ACS的URL，没有配置时显示为Null
ACS username	CPE连接ACS的用户名，没有配置时显示为Null
Periodic inform	周期发送Inform报文的开启情况： <ul style="list-style-type: none"> <li>Enabled: 表示已开启</li> <li>Disabled: 表示未开启</li> </ul>
Inform interval	发送Inform报文的周期，没有配置时显示为600s
Inform time	定期发送Inform报文的日期和时间，没有配置时显示为None
Wait timeout	无数据传输超时的时间
Connection retries	CWMP连接失败后自动重新连接的次数，没有配置时显示为Unlimited
Source IP interface	CPE上用于连接ACS的接口，没有配置时显示为None
STUN state	NAT穿越功能的开启状态： <ul style="list-style-type: none"> <li>Enabled: 表示已开启</li> <li>Disabled: 表示未开启</li> </ul>
SSL policy name	连接ACS采用的SSL策略名，没有配置时显示为Null

## 【相关命令】

- `display cwmp status`

### 1.1.20 display cwmp status

`display cwmp status` 命令用来显示 CWMP 的状态信息。

## 【命令】

`display cwmp status`

## 【视图】

任意视图

## 【缺省用户角色】

network-admin  
network-operator

## 【举例】

# 显示 CWMP 的状态信息。

```
<sysname> display cwmp status
CWMP state                               : Enabled
ACS URL of most recent connection        : http://www.acs.com:9090
ACS information source                    : User
```



```

ACS username of most recent connection      : newname
Connection status                          : Disconnected
Data transfer status                       : None
Most recent successful connection attempt   : None
Length of time before next connection attempt : 1096832s

```

表1-2 display cwmp status 命令显示信息描述表

字段	描述
CWMP state	CWMP的开启状态： <ul style="list-style-type: none"> <li>Enabled: 表示已开启</li> <li>Disabled: 表示未开启</li> </ul>
ACS URL of most recent connection	最近一次CPE使用的连接ACS的URL，没有配置时显示为Null
ACS information source	CPE获得ACS URL的方式，没有配置ACS URL时显示为None <ul style="list-style-type: none"> <li>User: 表示 ACS URL 为命令行配置或者 ACS 配置</li> <li>DHCP: 表示 ACS URL 为 DHCP 下发</li> <li>Default: 表示 ACS URL 为缺省配置</li> </ul>
ACS username of most recent connection	最近一次CPE使用的连接ACS的用户名，没有配置时显示为Null
Connection status	CPE的连接状态，包含： <ul style="list-style-type: none"> <li>Connected: 表示连接已建立</li> <li>Disconnected: 表示没有建立连接</li> <li>Waiting response: 表示正在等待响应报文</li> </ul>
Data transfer status	CPE的数据传输的状态，包含： <ul style="list-style-type: none"> <li>Uploading: 表示正在上传数据</li> <li>Downloading: 表示正在下载数据</li> <li>None: 表示没有数据在传输</li> </ul>
Most recent successful connection attempt	最近一次CPE和ACS成功连接的时间，最近没有成功连接时显示为None
Length of time before next connection attempt	距离下一次发起连接的时间，单位为秒。若CPE设备未检测到触发会话的事件，则显示为none

## 【相关命令】

- display cwmp configuration**

### 1.1.21 ssl client-policy

**ssl client-policy** 命令用来绑定 SSL 客户端策略。

**undo ssl client-policy** 命令用来恢复缺省情况。

## 【命令】

**ssl client-policy** *policy-name*

**undo ssl client-policy**

### 【缺省情况】

CWMP 没有绑定 SSL 客户端策略。

### 【视图】

CWMP 视图

### 【缺省用户角色】

network-admin

### 【参数】

*policy-name*: SSL 客户端策略名，为 1~31 个字符的字符串，不区分大小写。

### 【使用指导】

CWMP 是基于 HTTP/HTTPS 协议的，CWMP 报文作为 HTTP/HTTPS 报文的数据部分封装在 HTTP/HTTPS 报文中。如果 ACS 的 URL 以 http://开头，则使用 HTTP 协议，如果 ACS 的 URL 以 https://开头，则使用 HTTPS 协议。

使用 HTTPS 协议时，为了对 ACS 身份进行认证，需要绑定 SSL 客户端策略。关于 SSL 客户端策略的详细介绍和配置请参见“安全配置指导”中的“SSL”。

### 【举例】

# 设置 CWMP 引用的 SSL 客户端策略为 test。

```
<Sysname> system
```

```
[Sysname] cwmp
```

```
[Sysname-cwmp] ssl client-policy test
```

# 目 录

1 EAA.....	1-1
1.1 EAA配置命令.....	1-1
1.1.1 action cli .....	1-1
1.1.2 action reboot .....	1-2
1.1.3 action switchover .....	1-2
1.1.4 action syslog.....	1-3
1.1.5 commit .....	1-4
1.1.6 display rtm environment .....	1-5
1.1.7 display rtm policy .....	1-5
1.1.8 event cli.....	1-7
1.1.9 event hotplug.....	1-8
1.1.10 event interface.....	1-9
1.1.11 event process.....	1-11
1.1.12 event snmp oid.....	1-12
1.1.13 event snmp-notification .....	1-13
1.1.14 event syslog.....	1-14
1.1.15 event track .....	1-16
1.1.16 rtm cli-policy.....	1-18
1.1.17 rtm environment.....	1-18
1.1.18 rtm event syslog buffer-size.....	1-20
1.1.19 rtm scheduler suspend .....	1-20
1.1.20 rtm tcl-policy .....	1-21
1.1.21 running-time.....	1-22
1.1.22 user-role .....	1-22

# 1 EAA

## 1.1 EAA配置命令

### 1.1.1 action cli

**action cli** 命令用来配置事件发生时执行指定的命令行。

**undo action** 命令用来取消指定的操作。

#### 【命令】

```
action number cli command-line
undo action number
```

#### 【缺省情况】

监控策略下未配置 CLI 动作。

#### 【视图】

CLI 监控策略视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**number**: 动作序号，取值范围为 0~231。

**cli command-line**: 需要执行的命令。该参数可以是命令的不完整形式，比如为命令 **interface loopback 1** 的缩写形式 **int loop 1**，但需要用户保证其为设备可识别的合法命令，否则，该动作不能成功执行。

#### 【使用指导】

同一个监控策略下可以配置多个动作，当监控策略被触发后，系统会按照动作序号从小到大依次执行这些动作。当新增动作时，请确保动作的执行顺序是正确的。如果用户配置了相同序号的动作，当管理员执行 **commit** 命令时，最后一次执行的 **action** 生效。

如果本命令指定的命令行不在用户视图下，则必须先配置进入相应视图的 **action cli**，且进视图的 **action** 的编号应小于执行指定命令的 **action** 的编号。比如，要使用 CLI 策略来关闭接口 GigabitEthernet1/0/1，则需要配置三条 **action** 命令，**action 1 cli system-view**、**action 2 cli interface gigabitethernet 1/0/1**、**action 3 cli shutdown**。

配置 **action** 命令时，用户可以直接输入参数的值，也可以引用环境变量。关于环境变量的详细介绍请参见“网络管理和监控配置指导”中的“EAA”。

#### 【举例】

# 为 CLI 监控策略 **test** 配置动作：当 **test** 被触发时，关闭接口 GigabitEthernet1/0/1。

```
<Sysname> system-view
[Sysname] rtm cli-policy test
[Sysname-rtm-test] action 1 cli system-view
```

```
[Sysname-rtm-test] action 2 cli interface gigabitethernet 1/0/1
[Sysname-rtm-test] action 3 cli shutdown
```

### 1.1.2 action reboot

**action reboot** 命令用来配置事件发生时执行重启操作。

**undo action** 命令用来取消指定的操作。

#### 【命令】

```
action number reboot [ slot slot-number ]
undo action number
```

#### 【缺省情况】

监控策略下未配置重启动作。

#### 【视图】

CLI 监控策略视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**number**: 动作序号，取值范围为 0~231。

**slot slot-number**: 表示设备在 IRF 中的成员编号。不指定该参数时，表示重启 IRF 系统。

#### 【使用指导】

使用 **action reboot** 命令, 或者使用 **action cli** 命令并将 *command-line* 参数指定为 **reboot** 命令, 均可实现在事件发生时执行重启操作。只是 **action reboot** 命令会直接执行重启操作, 使用 **action cli** 命令时, 用户可以选择是否先保存当前配置, 再执行重启操作。

同一个监控策略下可以配置多个动作, 当监控策略被触发后, 系统会按照动作序号从小到大依次执行这些动作。当新增动作时, 请确保动作的执行顺序是正确的。如果用户配置了相同序号的动作, 当管理员执行 **commit** 命令时, 最后一次执行的 **action** 生效。

配置 **action** 命令时, 用户可以直接输入参数的值, 也可以引用环境变量。关于环境变量的详细介绍请参见“网络管理和监控配置指导”中的“EAA”。

#### 【举例】

# 为 CLI 监控策略 test 配置动作: 当 test 被触发时, 重启指定 slot。

```
<Sysname> system-view
[Sysname] rtm cli-policy test
[Sysname-rtm-test] action 3 reboot slot 1
```

### 1.1.3 action switchover

**action switchover** 命令用来配置事件发生时启动主备倒换。

**undo action** 命令用来取消指定的操作。

#### 【命令】

```
action number switchover
```

**undo action number**

【缺省情况】

监控策略下未配置主备倒换动作。

【视图】

CLI 监控策略视图

【缺省用户角色】

network-admin

【参数】

**number**: 动作序号，取值范围为 0~231。

【使用指导】

同一个监控策略下可以配置多个动作，当监控策略被触发后，系统会按照动作序号从小到大依次执行这些动作。当新增动作时，请确保动作的执行顺序是正确的。如果用户配置了相同序号的动作，当管理员执行 **commit** 命令时，最后一次执行的 **action** 生效。

即使当前设备不是主备环境（未部署从设备或者从设备未正常启动），该命令也会执行成功，但不会触发主设备和备设备的倒换动作。

【举例】

# 为 CLI 监控策略 **test** 配置动作：当 **test** 被触发时，执行主备倒换。

```
<Sysname> system-view
[Sysname] rtm cli-policy test
[Sysname-rtm-test] action 3 switchover
```

#### 1.1.4 action syslog

**action syslog** 命令用来配置事件发生时生成一条指定内容的日志。

**undo action** 命令用来取消指定的操作。

【命令】

**action number syslog priority priority facility local-number msg msg-body**  
**undo action number**

【缺省情况】

监控策略下未配置日志动作。

【视图】

CLI 监控策略视图

【缺省用户角色】

network-admin

【参数】

**number**: 动作序号，取值范围为 0~231。

**priority priority**: 生成的日志的优先级，取值范围为 0~7。优先级的值越小，优先级越高。

**facility local-number**: 生成日志的设备号，取值范围为 local0~local7。该参数主要用于在日志主机端标识不同的日志来源，以便查找和过滤不同日志源的日志。

**msg msg-body**: 生成的日志内容。

#### 【使用指导】

事件触发后生成的日志信息会交给信息中心模块，信息中心模块的配置将决定日志信息的发送规则和发送方向。关于信息中心的详细描述请参见“网络管理和监控配置指导”中的“信息中心”。

同一个监控策略下可以配置多个动作，当监控策略被触发后，系统会按照动作序号从小到大依次执行这些动作。当新增动作时，请确保动作的执行顺序是正确的。如果用户配置了相同序号的动作，当管理员执行 **commit** 命令时，最后一次执行的 **action** 生效。

配置 **action** 命令时，用户可以直接输入参数的值，也可以引用环境变量。关于环境变量的详细介绍请参见“网络管理和监控配置指导”中的“EAA”。

#### 【举例】

# 为 CLI 监控策略 **test** 配置动作：当 **test** 被触发时，生成一条优先级为 7、设备号为 local3、内容为 **hello** 的日志。

```
<Sysname> system-view
[Sysname] rtm cli-policy test
[Sysname-rtm-test] action 3 syslog priority 7 facility local3 msg hello
```

### 1.1.5 commit

**commit** 命令用来启用 CLI 监控策略。

#### 【命令】

**commit**

#### 【缺省情况】

CLI 监控策略未被启用。

#### 【视图】

CLI 监控策略视图

#### 【缺省用户角色】

network-admin

#### 【使用指导】

CLI 监控策略创建并配置事件和动作后，并不会立即生效，需要执行 **commit** 命令才会生效。

修改 CLI 监控策略的配置后，也需要执行 **commit** 命令，新配置才会生效。

#### 【举例】

# 启用 CLI 监控策略 **test**。

```
<Sysname> system-view
[Sysname] rtm cli-policy test
[Sysname-rtm-test] commit
```

## 1.1.6 display rtm environment

**display rtm environment** 命令用来显示用户自定义的 EAA 环境变量配置。

### 【命令】

**display rtm environment** [ *var-name* ]

### 【视图】

任意视图

### 【缺省用户角色】

network-admin  
network-operator

### 【参数】

**var-name**: 环境变量的名称，为 1~63 个字符的字符串，只能包含数字、字母和下划线，并且不能以下划线开头，区分大小写。不指定该参数时，显示所有环境变量的配置。

### 【举例】

```
# 显示用户自定义的所有 EAA 环境变量配置。
<Sysname> display rtm environment
Name                Value
save_cmd            save main force
show_run_cmd        display current-configuration
```

表1-1 display rtm environment 命令信息描述

主要字段	描述
Name	环境变量的名称，最多可显示30个字符。如果环境变量的名称超过30个字符，可使用 <b>display current-configuration</b> 命令查看
Value	环境变量的值，最多可显示30个字符。如果环境变量的值超过30个字符，可使用 <b>display current-configuration</b> 命令查看

## 1.1.7 display rtm policy

**display rtm policy** 命令用来显示监控策略的相关信息。

### 【命令】

**display rtm policy** { **active** | **registered** [ **verbose** ] } [ *policy-name* ]

### 【视图】

任意视图

### 【缺省用户角色】

network-admin  
network-operator



## 【参数】

**active:** 显示正在执行 **action** 命令的监控策略的相关信息。

**registered:** 显示已创建的监控策略的相关信息。

**verbose:** 显示监控策略的详细信息。

**policy-name:** 显示指定监控策略的相关信息。*policy-name* 表示监控策略的名称，为 1~63 个字符的字符串，区分大小写。不指定该参数时，显示所有正在执行的或者是已创建的监控策略的相关信息。

## 【使用指导】

本命令用来显示监控策略的相关信息，CLI 监控策略下生效的具体配置请在任意视图执行 **display current-configuration** 命令，或者在 CLI 监控策略视图下使用 **display this** 命令来查看。

## 【举例】

# 显示所有正在执行 **action** 命令的监控策略的相关信息。

```
<Sysname> display rtm policy active
JID   Type   Event      TimeActive      PolicyName
507   CLI    INTERFACE  Aug 29 14:55:55 2013 test
```

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

主要字段	描述
JID	任务ID，只有执行 <b>display rtm policy active</b> 命令时才显示该信息
Type	监控策略的类型，其中： <ul style="list-style-type: none"><li>TCL 表示这个策略是通过 Tcl 脚本定义的</li><li>CLI 表示这个策略是通过命令行定义的</li></ul>
Event	监控策略触发事件的类型，包括CLI、HOTPLUG、INTERFACE、PROCESS、SNMP、SNMP_NOTIF、SYSLOG和TRACK
TimeActive	监控策略开始运行的时间
PolicyName	监控策略的名称

# 显示所有监控策略的简要信息。

```
<Sysname> display rtm policy registered
Total number: 1
Type   Event      TimeRegistered      PolicyName
CLI    Aug 29 14:54:50 2017 test
```

表1-3 display rtm policy registered 命令显示信息描述表

主要字段	描述
Total number	监控策略的总个数
Type	监控策略的类型，其中： <ul style="list-style-type: none"><li>TCL 表示这个策略是通过 Tcl 脚本定义的</li><li>CLI 表示这个策略是通过命令行定义的</li></ul>

主要字段	描述
Event	监控策略触发事件的类型，包括CLI、HOTPLUG、INTERFACE、PROCESS、SNMP、SNMP_NOTIF、SYSLOG和TRACK
TimeRegistered	监控策略的创建时间
PolicyName	监控策略的名称

# 显示所有监控策略的详细信息。

```
<Sysname> display rtm policy registered verbose
Total number: 1

Policy Name: test
Policy Type: CLI
Event Type:
TimeRegistered: Aug 29 14:54:50 2017
User-role: network-operator
          network-admin
```

表1-4 display rtm policy registered verbose 命令显示信息描述表

主要字段	描述
Total number	监控策略的总数
Policy Name	监控策略的名称
Policy Type	监控策略的类型，其中： <ul style="list-style-type: none"> <li>TCL 表示这个策略是通过 Tcl 脚本定义的</li> <li>CLI 表示这个策略是通过命令行定义的</li> </ul>
Event Type	监控策略触发事件的类型，包括CLI、HOTPLUG、INTERFACE、PROCESS、SNMP、SNMP_NOTIF、SYSLOG和TRACK
TimeRegistered	监控策略的创建时间
User-role	执行监控策略的角色

### 1.1.8 event cli

**event cli** 命令用来配置命令行事件。

**undo event** 命令用来取消事件配置。

#### 【命令】

```
event cli { async [ skip ] | sync } mode { execute | help | tab } pattern
regular-exp
undo event
```

#### 【缺省情况】

未配置命令行事件。

## 【视图】

CLI 监控策略视图

## 【缺省用户角色】

network-admin

## 【参数】

**async** [**skip**]: 异步监控。如果指定 **skip** 参数, 则表示事件发生时, 只执行 CLI 监控策略中的动作, 不执行 **event cli** 中指定的命令; 如果不指定 **skip** 参数, 则表示事件发生时, CLI 监控策略和 **event cli** 中指定的命令同时执行。

**sync**: 同步监控。只有 CLI 监控策略执行成功, **event cli** 中指定的命令才能执行。

**mode** { **execute** | **help** | **tab** }: 对命令的监控方式。其中:

- **execute** 表示监控命令行的执行。当用户执行特定命令时, 触发 CLI 监控策略。
- **help** 表示监控命令行的“?”帮助。当用户对指定命令执行帮助操作时, 触发 CLI 监控策略。
- **tab** 表示监控命令行的 Tab 补全。当用户执行指定命令并使用 Tab 键自动补全功能时, 触发 CLI 监控策略。

**pattern** *regular-exp*: 用于匹配命令行的正则表达式。只要输入的命令中包含该字符串, 则匹配成功, 触发策略执行。关于正则表达式的详细描述请参见“基础配置指导”中的“CLI”。

## 【使用指导】

配置该事件后, 当用户输入指定的命令并执行相应动作(执行、帮助或者补全)就会触发策略执行。

同一监控策略下, 只能配置一个事件。如果多次执行 **event** 命令配置了不同事件, 最后一次配置并 **commit** 的生效。

## 【举例】

# 为 CLI 监控策略 **test** 配置异步 CLI 事件, 当用户输入命令中含有 **display interface brief** 字符并执行该命令时触发策略执行, 同时跳过命令行执行。

```
<Sysname>system-view
[Sysname] rtm cli-policy test
[Sysname-rmt-test] event cli async skip mode execute pattern display interface brief
```

# 为 CLI 监控策略 **test** 配置异步 CLI 事件, 当用户输入的命令中含有 **display interface brief** 字符并使用了<Tab>键补全功能时, 触发策略执行, 同时将<Tab>键补全的结果返回给用户。

```
<Sysname> system-view
[Sysname] rtm cli-policy test
[Sysname-rmt-test] event cli async mode tab pattern display interface brief
```

# 为 CLI 监控策略 **test** 配置同步 CLI 事件, 当用户输入的命令中含有 **display interface brief** 字符并使用了帮助功能时, 触发策略执行。系统等策略执行成功后, 返回帮助的结果。

```
<Sysname>system-view
[Sysname] rtm cli-policy test
[Sysname-rmt-test] event cli sync mode help pattern display interface brief
```

### 1.1.9 event hotplug

**event hotplug** 命令用来配置热插拔事件。

**undo event** 命令用来取消事件配置。

### 【命令】

```
event hotplug [ insert | remove ] slot slot-number  
undo event
```

### 【缺省情况】

未配置热插拔事件。

### 【视图】

CLI 监控策略视图

### 【缺省用户角色】

network-admin

### 【参数】

**insert**: 表示监控成员设备加入 IRF 事件。  
**remove**: 表示监控成员设备离开 IRF 事件。  
**slot slot-number**: 表示设备在 IRF 中的成员编号。

### 【使用指导】

配置该事件后，当指定成员设备加入/离开 IRF，会触发监控策略执行。  
不指定 **insert** 和 **remove** 参数时，表示同时监控加入和离开事件。  
同一监控策略下，只能配置一个事件。如果多次执行 **event** 命令配置了不同事件，最后一次配置并 **commit** 的生效。

### 【举例】

# 为 CLI 监控策略 **test** 配置监控事件：指定成员设备加入或离开时触发策略执行。

```
<Sysname> system-view  
[Sysname] rtm cli-policy test  
[Sysname-rtm-test] event hotplug slot 1
```

## 1.1.10 event interface

**event interface** 命令用来配置接口事件。  
**undo event** 命令用来取消事件配置。

### 【命令】

```
event interface interface-type interface-number monitor-obj monitor-obj  
start-op start-op start-val start-val restart-op restart-op restart-val  
restart-val [ interval interval ]  
undo event
```

### 【缺省情况】

未配置接口事件。

### 【视图】

CLI 监控策略视图

【缺省用户角色】

network-admin

【参数】

*interface-type interface-number*: 表示要监控的接口类型和编号。

**monitor-obj** *monitor-obj*: 表示要监控的对象，具体描述请见 表 1-5。

**start-op** *start-op*: 触发监控策略执行的操作码，取值如 表 1-6 所示。

**start-val** *start-val*: 触发监控策略执行的监控对象的值，取值范围为 0~4294967295，单位请见 表 1-5。

**restart-op** *restart-op*: 重新开启触发开关的操作码，具体描述请见 表 1-6。

**restart-val** *restart-val*: 重新开启触发开关的监控对象的值，取值范围为 0~4294967295，单位请见 表 1-5。

**interval** *interval*: 监控对象数据的采样周期，取值范围为 1~4294967295，单位为秒，缺省值为 300。

表1-5 监控对象说明

监控对象	含义
input-drops	采样周期内，接口接收方向丢弃包的个数
input-errors	采样周期内，接口接收到的错误包的个数
output-drops	采样周期内，接口发送方向丢弃包的个数
output-errors	采样周期内，接口发送出去的错误包的个数
rcv-bps	采样周期内，接口接收速率，单位为比特/秒
rcv-broadcasts	采样周期内，接口接收到的广播包个数
rcv-pps	接口接收速率，单位为包/秒
tx-bps	接口发送速率，单位为比特/秒
tx-pps	接口发送速率，单位为包/秒

表1-6 比较操作符说明

比较操作符	含义
eq	等于
ge	大于等于
gt	大于
le	小于等于
lt	小于
ne	不等于

## 【使用指导】

接口事件的触发过程为：

- (1) 配置该事件后，触发开关立即打开。
- (2) 当指定接口上的指定报文的数目达到 **start-op start-op start-val start-val** 参数指定的条件时，触发监控策略执行一次（第一次执行），并关闭触发开关，但系统会继续监控接口事件。
- (3) 当满足 **restart-op restart-op restart-val restart-val** 参数指定的条件时，才重新开启触发开关。
- (4) 如果指定接口上的指定报文的数目再次达到 **start-op start-op start-val start-val** 参数指定的条件时，则再次触发监控策略执行一次（第二次执行），并关闭触发开关，系统继续监控接口事件。
- (5) 如此循环。

同一监控策略下，只能配置一个事件。如果多次执行 **event** 命令配置了不同事件，最后一次配置并 **commit** 的生效。

## 【举例】

# 为 CLI 监控策略 **test** 配置监控事件：每 60 秒获取一次 GigabitEthernet1/0/1 接收到的错误包的个数。当错误包的个数大于 1000 时触发 **test** 执行并关闭触发开关，当错误包的个数小于 50 时重新开启触发开关，如此循环。

```
<Sysname> system-view
[Sysname] rtm cli-policy test
[Sysname-rtm-test] event interface gigabitethernet 1/0/1 monitor-obj input-errors start-op
gt start-val 1000 restart-op lt restart-val 50 interval 60
```

### 1.1.11 event process

**event process** 命令用来配置进程事件。

**undo event** 命令用来取消事件配置。

## 【命令】

```
event process { exception | restart | shutdown | start } [ name process-name
[ instance instance-id ] ] [ slot slot-number ]
undo event
```

## 【缺省情况】

未配置进程事件。

## 【视图】

CLI 监控策略视图

## 【缺省用户角色】

network-admin

## 【参数】

**exception**：监控进程异常事件。当进程发生异常时，触发 CLI 监控策略。

**restart**: 监控进程重启事件。当进程重启时，触发 CLI 监控策略。

**shutdown**: 监控进程关闭事件。当进程关闭时，触发 CLI 监控策略。

**start**: 监控进程启动事件。当进程启动时，触发 CLI 监控策略。

**name process-name**: 监控的用户态进程的名称，可以是当前正在运行的进程也可以是没有运行的进程。不指定该参数时，表示所有用户态进程。

**instance instance-id**: 监控的用户态进程的实例的编号，取值范围为 0~4294967295，可以是当前不存在的实例号。不指定该参数时，表示指定进程的所有实例。

**slot slot-number**: 表示设备在 IRF 中的成员编号。不指定该参数时，表示所有成员设备。

### 【使用指导】

配置该事件后，当指定进程异常、关闭、启动或重启时（可以为用户命令行触发的或者系统自动触发的），均触发监控策略执行。

同一监控策略下，只能配置一个事件。如果多次执行 **event** 命令配置了不同事件，最后一次配置并 **commit** 的生效。

### 【举例】

# 为 CLI 监控策略 **test** 配置监控事件：当进程 **snmpd** 重启时触发策略执行。

```
<Sysname>system-view
[Sysname] rtm cli-policy test
[Sysname-rtm-test] event process restart name snmpd
```

## 1.1.12 event snmp oid

**event snmp oid** 命令用来配置 SNMP 操作事件。

**undo event** 命令用来取消事件配置。

### 【命令】

```
event snmp oid oid monitor-obj { get | next } start-op start-op start-val
start-val restart-op restart-op restart-val restart-val [ interval
interval ]
undo event
```

### 【缺省情况】

未配置 SNMP 操作事件。

### 【视图】

CLI 监控策略视图

### 【缺省用户角色】

network-admin

### 【参数】

**oid oid**: 表示需要监控的 MIB 对象的 OID，为 1~256 个字符的字符串。

**monitor-obj { get | next }**: 表示需要监控的 SNMP 操作。**get** 表示 SNMP Get 操作；**next** 表示 SNMP Get Next 操作。

**start-op start-op**: 触发监控策略执行的操作码，取值如 [表 1-6](#) 所示。

**start-val start-val**: 触发监控策略执行的值。可以是 SNMP 模块支持的所有类型，例如，数字、字符串等。为 1~512 个字符的字符串。如果该值中包含空格，需要在值的首末添加英文格式的引号，形如"xxx xxx"。

**restart-op op**: 重新开启触发开关的操作码，取值如 表 1-6 所示。

**restart-val restart-val**: 重新开启触发开关的监控对象的值。可以是 SNMP 模块支持的所有类型，例如，数字、字符串等。为 1~512 个字符的字符串。如果该值中包含空格，需要在值的首末添加英文格式的引号，形如"xxx xxx"。

**interval interval**: 获取监控对象数据的时间间隔，取值范围为 1~4294967295，单位为秒，缺省值为 300。

### 【使用指导】

SNMP 操作事件的触发过程为：

- (1) 配置该事件后，触发开关立即打开。
- (2) 此后系统按照用户设定的 **interval interval** 定时获取设备上某个 **OID** 对应节点的值，该值达到 **start-op start-op start-val start-val** 参数指定的条件时，触发监控策略执行一次（第一次执行），并关闭触发开关，但系统会继续监控 SNMP 操作事件。
- (3) 当满足 **restart-op restart-op restart-val restart-val** 参数指定的条件时，才重新开启触发开关。
- (4) 如果用户获取的 MIB 对象的值再次达到 **start-op start-op start-val start-val** 参数指定的条件时，则再次触发监控策略执行一次（第二次执行），并关闭触发开关，系统继续监控 SNMP 操作事件。
- (5) 如此循环。

同一监控策略下，只能配置一个事件。如果多次执行 **event** 命令配置了不同事件，最后一次配置并 **commit** 的生效。

配置该命令前，请先开启 SNMP 功能，否则，该命令配置失败。

### 【举例】

# 为 CLI 监控策略 test 配置监控事件：系统每 5 秒检查 MIB 对象 1.3.6.4.9.9.42.1.2.1.6.4 的值，当该值等于 1 时触发执行监控策略并关闭监控开关，当等于 2 时重新启动监控。

```
<Sysname> system-view
[Sysname] rtm cli-policy test
[Sysname-rtm-test] event snmp oid 1.3.6.4.9.9.42.1.2.1.6.4 monitor-obj get start-op eq
start-val 1 restart-op eq restart-val 2 interval 5
```

## 1.1.13 event snmp-notification

**event snmp-notification** 命令用来配置 SNMP 告警事件。

**undo event** 命令用来取消事件配置。

### 【命令】

```
event snmp-notification oid oid oid-val oid-val op op [ drop ]
undo event
```



### 【缺省情况】

未配置 SNMP 告警触发事件。

### 【视图】

CLI 监控策略视图

### 【缺省用户角色】

network-admin

### 【参数】

**oid** *oid*: 告警信息中携带的 MIB 对象的 OID，为 1~256 个字符的字符串。

**oid-val** *oid-val*: 告警信息中携带的 MIB 对象的值。可以是 SNMP 模块支持的所有类型，例如，数字、字符串等。为 1~512 个字符的字符串。如果该值中包含空格，需要在值的首末添加英文格式的引号，形如“xxx xxx”。

**op** *op*: 比较操作码，取值如 [表 1-6](#) 所示。

**drop**: 表示匹配成功后丢弃该告警信息。不指定该参数时，表示正常发送。

### 【使用指导】

配置该事件后，当系统生成一条告警，且告警中携带的 MIB 对象（由 *oid* 参数指定）的值到达 **oid-val** *oid-val* **op** *op* 指定的条件时，触发监控策略执行。

同一监控策略下，只能配置一个事件。如果多次执行 **event** 命令配置了不同事件，最后一次配置并 **commit** 的生效。

配置该命令前，请先开启 SNMP 功能，否则，该命令配置失败。

### 【举例】

# 为 CLI 监控策略 **test** 配置 SNMP 告警监控事件：当告警信息中包含的登录用户名为 **admin** 时，触发策略且丢弃该告警。（登录用户名对应的 MIB 节点的 OID 为 1.3.6.1.4.1.25506.2.2.1.1.2.1.0）

```
<Sysname> system-view
[Sysname] rtm cli-policy test
[Sysname-rtm-test] event snmp-notification oid 1.3.6.1.4.1.25506.2.2.1.1.2.1.0 oid-val
admin op eq drop
```

## 1.1.14 event syslog

**event syslog** 命令用来配置日志事件。

**undo event** 命令用来取消事件配置。

### 【命令】

```
event syslog priority { priority | all } msg msg occurs times period period
undo event
```

### 【缺省情况】

未配置日志事件。

### 【视图】

CLI 监控策略视图

【缺省用户角色】

network-admin

【参数】

**priority** { *priority* | **all** }：表示需要匹配的日志的优先级。其中：

- *priority* 表示需要匹配的日志的最低优先级，取值范围为 0~7。优先级的数值越小，优先级越高。当 *level* 配置为 3 时，表示能匹配优先级为 0~3 的日志。
- **all** 表示和任意优先级匹配，效果同 *level* 配置为 7。

**msg** *msg*：表示需要匹配的日志的部分或全部。*msg* 表示包含日志部分或全部内容的正则表达式，为 1~255 个字符的字符串。

**occurs** *times* **period** *period*：表示设备在 *period* 时间内收到 *times* 次指定日志时触发监控策略执行。*times* 的取值范围为 1~32；*period* 的取值范围为 1~4294967295，单位为秒。

【使用指导】

配置该事件后，当系统在指定时间段内生成指定的日志信息时触发监控策略执行。为了防止循环触发，RTM 模块产生的日志不会触发策略。

同一监控策略下，只能配置一个事件。如果多次执行 **event** 命令配置了不同事件，最后一次配置并 **commit** 的生效。

正则表达式支持多种特殊字符，特殊字符的匹配规则如 [表 1-7](#) 所示。

表1-7 正则表达式中的特殊字符描述表

特殊字符	含义	举例
^	匹配以指定字符开始的行	^u只能匹配以u开始的行，不能匹配以Au开始的行
\$	匹配以指定字符结束的行	u\$只能匹配以u结尾的行，不能匹配以uA结尾的行
.	通配符，可代表任何一个字符	.s可以匹配as和bs等
*	匹配星号前面的字符或字符串零次或多次	<ul style="list-style-type: none"><li>• zo*可以匹配 z 以及 zoo</li><li>• (zo)*可以匹配 zo 以及 zozo</li></ul>
+	匹配+前面的字符或字符串一次或多次	zo+可以匹配zo以及zoo，但不能匹配z
	匹配 左边或右边的整个字符串	def int只能匹配包含def或者int的字符串所在的行
()	表示字符串，一般与“+”或“*”等符号一起使用	(123A)表示字符串123A；408(12)+可以匹配40812或408121212等字符串，但不能匹配408
\index	表示重复一次指定字符串，字符串是指\前用()括起来的字符串，index对应\前字符串的顺序号按从左至右的顺序从1开始编号：如果\前面只有一个字符串，则index只能为1；如果\前面有n个字符串，则index可以为1到n中的任意整数	(string)\1表示把string重复一次，匹配的字符串必须包含stringstring；(string1)(string2)\2表示把string2重复一次，匹配的字符串必须包含string1string2string2；(string1)(string2)\1\2表示先把string1重复一次，再重复一次string2，匹配的字符串必须包含string1string2string1string2

特殊字符	含义	举例
[ ]	表示字符选择范围，将以选择范围内的单个字符为条件进行匹配，只要字符串里包含该范围的某个字符就能匹配到	<ul style="list-style-type: none"> <li>• [16A]表示可以匹配到的字符串只需要包含 1、6 或 A 中任意一个</li> <li>• [1-36A] 表示可以匹配到的字符串只需要包含 1、2、3、6 或 A 中任意一个（-为连接符）</li> </ul> <p>如果[]需要作为普通字符出现在[]内时，必须把[]写在[]中字符的最前面，形如[]string]，才能匹配到]。[]没有这样的限制</p>
[^ ]	表示选择范围外的字符，将以单个字符为条件进行匹配，只要字符串里包含该范围外的某个字符就能匹配到	[^16A]表示可匹配的字符串只需要包含1、6和A之外的任意字符，该字符串也可以包含字符1、6或A，但不能只包含这三个字符。例如[^16A]可以匹配abc、m16，不能匹配1、16、16A
{n}	n是一个非负整数，匹配n次	o{2}不能匹配Bob，但是能匹配food
{n,}	n是一个非负整数，至少匹配n次	o{2,}不能匹配Bob，但能匹配foooooo
{n,m}	m和n均为非负整数，其中n小于等于m。只要字符串里包含n到m个某字符就能匹配到	o{1,3}能匹配fod、food、foood、foooooo，但不能匹配fd
\<	匹配包含指定字符串的字符串，字符串前面如果有字符则不能是数字、字母和下划线	\<do匹配单词domain，还可以匹配字符串doa
\>	匹配包含指定字符串的字符串，字符串后面如果有字符则不能是数字、字母和下划线	do\>匹配单词undo，还可以匹配字符串cdo
\b	匹配一个单词边界，也就是指单词和空格间的位置	er\b可以匹配never，但不能匹配verb \ber可以匹配erase，但不能匹配verb
\B	匹配非单词边界	er\B能匹配verb，但不能匹配never
\w	\w等效于[A-Za-z0-9_]，是数字、字母或下划线	\w能匹配vlan，\W还能匹配service
\W	\W等效于[^A-Za-z0-9_]，是除了数字、字母和下划线之外的任意字符	\Wa可以匹配-a，但是不能匹配2a和ba等
\	转义操作符，\后紧跟本表中罗列的单个特殊字符时，将去除特殊字符的特定含义	<ul style="list-style-type: none"> <li>• \\可以匹配包含\的字符串</li> <li>• \^可以匹配包含^的字符串</li> <li>• \\b 可以匹配包含\b 的字符串</li> </ul>

### 【举例】

# 为 CLI 监控策略 test 配置监控事件：当优先级高于或等于 3、内容中含有 down 的日志在 6 秒内出现过 5 次时触发执行策略。

```
<Sysname> system-view
[Sysname] rtm cli-policy test
[Sysname-rtm-test] event syslog priority 3 msg down occurs 5 period 6
```

#### 1.1.15 event track

event track 命令用来配置 Track 事件。

**undo event** 命令用来取消事件配置。

#### 【命令】

```
event track track-list state { negative | positive } [ suppress-time  
suppress-time ]  
undo event
```

#### 【缺省情况】

未配置 Track 事件。

#### 【视图】

CLI 监控策略视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**track-list**: Track 项列表，表示多个 Track 项的序号。表示方式为 **track-list = { track-entry-number [ to track-entry-number ] }&<1-16>**。其中，**track-entry-number** 为 Track 项的序号，取值范围为 1~1024。**&<1-16>**表示前面的参数最多可以输入 16 次。

**state { negative | positive }**: 监控 Track 项的状态变化。其中：

- **negative**: 表示当 Track 项由 Positive 状态变为 Negative 状态时，触发该事件。
- **positive**: 表示当 Track 项由 Negative 状态变为 Positive 状态时，触发该事件。

**suppress-time suppress-time**: 抑制时间，取值范围为 1~4294967295，单位为秒，缺省值为 0。

#### 【使用指导】

配置该事件后，当关联的 Track 项状态由 Positive 变为 Negative 或者 Negative 变为 Positive 时，触发监控策略执行。如果关联多个 Track 项，则最后一个处于 Positive（Negative）状态的 Track 项变为 Negative（Positive）时，触发监控策略执行。

如果配置了抑制时间，触发策略的同时开始计时，定时器超时前，收到状态从 Positive（Negative）变为 Negative（Positive）的消息，直接丢弃，不会处理。直到定时器超后，收到状态从 Positive（Negative）变为 Negative（Positive）的消息才处理，再一次触发策略执行。

同一监控策略下，只能配置一个事件。如果多次执行 **event** 命令配置了不同事件，则最新配置并 **commit** 的生效。

#### 【举例】

# 为 Track 监控策略 test 配置监控事件：监控 Track 项 1 到 8 全部变为 Negative 时，触发执行策略，并配置抑制时间为 180 秒。

```
<Sysname> system-view  
[Sysname] rtm cli-policy test  
[Sysname-rtm-test] event track 1 to 8 state negative suppress-time 180
```

### 1.1.16 rtm cli-policy

**rtm cli-policy** 命令用来创建 CLI 监控策略，并进入 CLI 监控策略视图。如果指定的 CLI 监控策略已经存在，则直接进入 CLI 监控策略视图。

**undo rtm cli-policy** 命令来删除指定的 CLI 监控策略。

#### 【命令】

```
rtm cli-policy policy-name
undo rtm cli-policy policy-name
```

#### 【缺省情况】

不存在 CLI 监控策略。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*policy-name*: CLI 监控策略的名称。为 1~63 个字符的字符串，区分大小写。

#### 【使用指导】

创建 CLI 监控策略后，用户可以通过命令行给 CLI 监控策略配置触发事件以及需要执行的动作。

CLI 监控策略配置完成后（即配置完触发事件以及需要执行的动作后），必须执行 **commit** 命令才能启用 CLI 监控策略，使配置的事件和动作生效。

多次执行该命令可以创建多个 CLI 监控策略且数量没有限制。请尽量确保同时启用的策略间动作不要冲突，因为当系统同时执行多个策略，且不同策略间动作有冲突时，执行结果是随机的。

Tcl 监控策略和 CLI 监控策略的名称可以相同。

#### 【举例】

# 创建 CLI 监控策略 test 并进入 CLI 监控策略视图。

```
<Sysname> system-view
[Sysname] rtm cli-policy test
```

#### 【相关命令】

- **commit**

### 1.1.17 rtm environment

**rtm environment** 命令用来创建监控策略的环境变量。

**undo rtm environment** 命令来删除指定的环境变量。

#### 【命令】

```
rtm environment var-name var-value
undo rtm environment var-name
```

【缺省情况】

不存在用户自定义的环境变量。系统中支持一系列内部环境变量，不同事件支持的内部环境变量及其意义有所不同，请参见 [表 1-8](#)。

表1-8 内部环境变量描述表

事件	内部环境变量的名称以及描述
CLI	_cmd: 匹配上的命令
SYSLOG	_syslog_pattern: 匹配的日志信息的内容
HOTPLUG	_slot: 加入或离开IRF的成员设备的编号
INTERFACE	_ifname: 接口的名称
SNMP	_oid: SNMP操作中携带的OID _oid_value: OID对应节点的值
SNMP TRAP	_oid: SNMP告警信息中携带的OID
PROCESS	_process_name: 进程的名称
公共环境变量	<ul style="list-style-type: none"><li>• _event_id: 事件的 ID</li><li>• _event_type: 事件的类型</li><li>• _event_type_string: 事件类型字符串，用于对事件类型进行详细描述</li><li>• _event_time: 事件发生的时间</li><li>• _event_severity: 事件的严重级别</li></ul>

【视图】

系统视图

【缺省用户角色】

network-admin

【参数】

**var-name**: 环境变量的名称，为 1~63 个字符的字符串，只能包含数字、字母和下划线，并且不能以下划线开头，区分大小写。

**var-value**: 环境变量的值。

【使用指导】

在配置监控策略的动作时，用户可以在应该输入参数的地方输入“\$环境变量名”，表示此处需要引用环境变量值。系统在运行监控策略的时候，会自动用环境变量值去替代“\$环境变量名”。

当同时用到多个环境变量时，请按照 \$variable\_name1\$variable\_name2...\$variable\_nameN 的格式输入。

【举例】

# 设置环境变量 if，其值为 interface。

```
<Sysname> system-view
[Sysname] rtm environment if interface
```

### 1.1.18 rtm event syslog buffer-size

**rtm event syslog buffer-size** 命令用来配置 EAA 监控的日志缓冲区的大小。

**undo rtm event syslog buffer-size** 命令用来恢复缺省情况。

#### 【命令】

```
rtm event syslog buffer-size buffer-size  
undo rtm event syslog buffer-size
```

#### 【缺省情况】

EAA 监控的日志缓冲区的大小为 50000 条。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*buffer-size*: EAA 监控的日志缓冲区的大小，取值范围为 1~500000，单位为条。

#### 【使用指导】

为 EAA 策略配置日志事件，并确认执行后，系统会将生成的日志同时复制一份放入 EAA 监控的日志缓冲区。EAA 模块将该缓冲区中的日志取出并和日志事件进行匹配，如果匹配成功，则执行动作；如果匹配失败，则不执行动作。

通常情况下，这个日志缓冲区的大小采用缺省值即可。当某特性功能异常或者用户开启了多个调试开关时，设备会在短时间内产生大量日志，导致一些日志来不及匹配便被丢弃，此时可根据当前设备内存的使用情况，适当增加 EAA 监控的日志缓冲区的大小。

#### 【举例】

# 配置 EAA 监控的日志缓冲区的大小为 1000 条。

```
<Sysname> system-view  
[Sysname] rtm event syslog buffer-size 1000
```

#### 【相关命令】

- **event syslog**

### 1.1.19 rtm scheduler suspend

**rtm scheduler suspend** 命令用来暂停运行所有的监控策略，包括 CLI 监控策略和 Tcl 监控策略。

**undo rtm scheduler suspend** 命令用来恢复运行监控策略。

#### 【命令】

```
rtm scheduler suspend  
undo rtm scheduler suspend
```

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【使用指导】

当监控策略被频繁触发影响用户使用设备或者用户需要修改 Tcl 监控策略的脚本内容时，可以使用本命令暂停运行所有的监控策略。

监控策略被暂停后，即便满足条件，也不再触发 **action** 命令。

如果执行本命令时，某个监控策略正在执行 **action** 命令，则这个监控策略会继续执行完所有的 **action** 命令，再被暂停。

## 【举例】

```
# 暂停所有的监控策略。
<Sysname> system-view
[Sysname] rtm scheduler suspend
```

### 1.1.20 rtm tcl-policy

**rtm tcl-policy** 命令用来创建并启用 Tcl 监控策略，并将该策略与 Tcl 脚本绑定。

**undo rtm tcl-policy** 命令来删除 Tcl 监控策略。

## 【命令】

```
rtm tcl-policy policy-name tcl-filename
undo rtm tcl-policy policy-name
```

## 【缺省情况】

不存在 Tcl 监控策略。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**policy-name**: Tcl 监控策略的名称。为 1~63 个字符的字符串，区分大小写。

**tcl-filename**: Tcl 脚本文件的名称。文件名区分大小写，扩展名必须为“.tcl”，扩展名不区分大小写，Tcl 脚本文件必须为设备存储介质上存在的文件。

## 【使用指导】

策略的具体内容由绑定的 Tcl 脚本来定义。脚本中会定义策略的触发事件、事件触发时要执行的操作、执行操作需要的角色、策略的运行时间等参数。

配置 Tcl 监控策略时，**tcl-filename** 请使用相对路径并确保所有的成员设备上都存在该脚本，以免主备倒换或脚本所在成员设备离开 IRF 后，策略无法执行。



Tcl 监控策略启用后，不允许修改 Tcl 脚本。如需修改，请先暂停 Tcl 监控策略，修改后，再启用 Tcl 监控策略。否则，Tcl 监控策略将不能运行。

不能通过重复执行本命令修改 Tcl 监控策略绑定的 Tcl 脚本。如需修改，请先通过 **undo rtm tcl-policy** 命令删除该 Tcl 监控策略，再重新配置。

#### 【举例】

# 创建并启用 Tcl 监控策略 test，并将它和脚本 test.tcl 绑定。

```
<Sysname> system-view
[Sysname] rtm tcl-policy test test.tcl
```

### 1.1.21 running-time

**running-time** 命令用来配置 CLI 监控策略执行动作的持续时间。

**undo running-time** 命令用来恢复缺省情况。

#### 【命令】

```
running-time time
undo running-time
```

#### 【缺省情况】

CLI 监控策略执行动作的持续时间为 20 秒。

#### 【视图】

CLI 监控策略视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**time**: CLI 监控策略执行动作的持续时间，取值范围为 0~31536000，单位为秒。取值为 0 时，表示策略可以永久运行下去。

#### 【使用指导】

当条件满足、监控策略被触发执行动作时，系统会开始计时。当运行时间到，即便策略还没有执行完动作，也会立即停止执行。该命令用于限制动作的执行时间，以免动作长时间执行占用系统资源。而策略是否会触发以及停止后是否会被再次触发则由 **event** 配置决定。

#### 【举例】

# 配置 CLI 监控策略 test 执行动作的持续时间为 60 秒。

```
<Sysname> system-view
[Sysname] rtm cli-policy test
[Sysname-rtm-test] running-time 60
```

### 1.1.22 user-role

**user-role** 命令用来配置执行 CLI 监控策略时使用的用户角色。

**undo user-role** 命令用来删除已经配置的指定用户角色。

### 【命令】

```
user-role role-name  
undo user-role role-name
```

### 【缺省情况】

执行 CLI 监控策略时使用的用户角色为创建该策略的用户的角色。

### 【视图】

CLI 监控策略视图

### 【缺省用户角色】

network-admin

### 【参数】

*role-name*: 用户角色名称，为 1~63 个字符的字符串，区分大小写。

### 【使用指导】

用户角色中定义了允许用户操作哪些系统功能以及资源对象，设备支持的每条命令都有缺省用户角色，如果监控策略中指定的用户角色权限比命令行的缺省用户角色的权限小，则不能执行该命令以及该命令后面的所有动作。如果指定的用户角色不存在，则监控策略不能执行。如果给某个监控策略配置了多个用户角色，则使用这些用户角色权限的并集去执行该策略。例如，给某策略配置了用户角色 A 和 B，如果策略中的动作是角色 A 或者 B 允许执行的，则策略可以执行；如果策略中存在角色 A 和 B 都不能执行的命令，则该命令以及该命令后面的所有动作都不能执行。关于用户角色的详细描述请参见“基础配置指导”中的“RBAC”。

同一监控策略下可配置多个用户角色，最多可以配置 64 个有效用户角色，超过该上限后，新配置的用户角色即便 **commit** 也不会生效。

安全日志管理员角色与其它用户角色互斥：配置安全日志管理员角色后，系统会自动删除已配置的其它用户角色；配置其它用户角色后，系统会自动删除已配置的安全日志管理员角色。

### 【举例】

# 配置执行 CLI 监控策略 test 时使用的用户角色为 network-admin 和 admin。

```
<Sysname> system-view  
[Sysname] rtm cli-policy test  
[Sysname-rtm-test] user-role network-admin  
[Sysname-rtm-test] user-role admin
```

# 目 录

1 进程监控和维护.....	1-1
1.1 进程监控和维护命令.....	1-1
1.1.1 display exception context .....	1-1
1.1.2 display exception filepath .....	1-5
1.1.3 display kernel deadlock .....	1-6
1.1.4 display kernel deadlock configuration.....	1-9
1.1.5 display kernel exception .....	1-10
1.1.6 display kernel reboot .....	1-13
1.1.7 display kernel starvation.....	1-16
1.1.8 display kernel starvation configuration.....	1-19
1.1.9 display process .....	1-20
1.1.10 display process cpu.....	1-23
1.1.11 display process log.....	1-24
1.1.12 display process memory .....	1-25
1.1.13 display process memory heap.....	1-26
1.1.14 display process memory heap address .....	1-28
1.1.15 display process memory heap size .....	1-29
1.1.16 exception filepath.....	1-30
1.1.17 monitor kernel deadlock action .....	1-31
1.1.18 monitor kernel deadlock enable .....	1-32
1.1.19 monitor kernel deadlock exclude-thread.....	1-33
1.1.20 monitor kernel deadlock time.....	1-34
1.1.21 monitor kernel starvation enable.....	1-34
1.1.22 monitor kernel starvation exclude-thread .....	1-35
1.1.23 monitor kernel starvation time .....	1-36
1.1.24 monitor process .....	1-37
1.1.25 monitor thread .....	1-42
1.1.26 process core .....	1-45
1.1.27 reset exception context .....	1-46
1.1.28 reset kernel deadlock .....	1-47
1.1.29 reset kernel exception .....	1-47
1.1.30 reset kernel reboot .....	1-48
1.1.31 reset kernel starvation.....	1-48



# 1 进程监控和维护

## 1.1 进程监控和维护命令

**display memory**、**display process**、**display process cpu**、**monitor process** 和 **monitor thread** 命令既可显示用户态进程的相关信息，又可显示内核线程的相关信息，为简便起见，在这些命令中，统一使用“进程”一词来指代。

### 1.1.1 display exception context

**display exception context** 命令用来显示用户态进程异常时的上下文信息。

#### 【命令】

```
display exception context [ count value ] [ slot slot-number [ cpu cpu-number ] ]
```

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**count** *value*: 表示上下文信息的显示个数，取值范围为 1~20，缺省值为 1。

**slot** *slot-number*: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。

**cpu** *cpu-number*: 表示 CPU 的编号。

#### 【使用指导】

当用户态进程发生一次异常，系统会生成一个 **core** 文件，还会生成一条上下文信息，用于记录异常用户态进程的 ID、生成 **core** 文件的时间、**core** 文件存放的位置、栈信息和寄存器信息。一个 **core** 文件对应一条上下文信息，最多可记录的上下文信息数和可记录的 **core** 文件数目相同。

#### 【举例】

# 显示在 x86 体系 32 位登录设备上的异常上下文信息。

```
<Sysname> display exception context
Index 1 of 1
-----
Crashed PID: 120 (routed)
Crash signal: SIGBUS
Crash time: Tue Apr 9 17:14:30 2013
Core file path:
flash:/core/node0_routed_120_7_20130409-171430_1365527670.core
#0  0xb7caba4a
#1  0x0804cb79
#2  0xb7cd77c4
```

#3 0x08049f45

Backtrace stopped.

```
Registers' content
eax:0xffffffffc ebx:0x00000003 ecx:0xbfe244ec edx:0x0000000a
esp:0xbfe244b8 ebp:0xbfe244c8 esi:0xfffffffff edi:0xbfe24674
eip:0xb7caba4a eflag:0x00000292 cs:0x00000073 ss:0x0000007b
ds:0x0000007b es:0x0000007b fs:0x00000000 gs:0x00000033
```

# 显示在 x86 体系 64 位登录设备上的异常上下文信息。

<Sysname> display exception context

Index 1 of 1

-----

Crashed PID: 121 (routed)

Crash signal: SIGBUS

Crash time: Sun Mar 31 11:12:21 2013

Core file path:

flash:/core/node0\_routed\_121\_7\_20130331-111221\_1364728341.core

#0 0x00007fae7dbad20c

#1 0x00000000004059fa

#2 0x00007fae7dbd96c0

#3 0x0000000000402b29

Backtrace stopped.

```
Registers' content
rax:0xffffffffffffffffc rbx:0x00007fff88a5dd10
rcx:0xfffffffffffffffff rdx:0x000000000000000a
rsi:0x00007fff88a5dd10 rdi:0x0000000000000003
rbp:0x00007fff88a5dcf0 rsp:0x00007fff88a5dcf0
r8:0x00007fae7ea587e0 r9:0x0000000000000079
r10:0xfffffffffffffffff r11:0x0000000000000246
r12:0x0000000000405b18 r13:0x00007fff88a5ff7a
r14:0x00007fff88a5de30 r15:0x0000000000000000
rip:0x00007fae7dbad20c flag:0x0000000000000246
cs:0x0000000000000033 ss:0x000000000000002b
ds:0x0000000000000000 es:0x0000000000000000
fs:0x0000000000000000 gs:0x0000000000000000
fs_base:0x00007fae80a5d6a0 gs_base:0x0000000000000000
orig_ax:0x00000000000000e8
```

# 显示在 powerpc 体系 32 位登录设备上的异常上下文信息。

<Sysname> display exception context

Index 1 of 1

-----

Crashed PID: 133 (routed)

Crash signal: SIGBUS

Crash time: Wed Apr 10 15:47:49 2013

Core file path:

flash:/core/node0\_routed\_133\_7\_20130410-154749\_1365608869.core

#0 0x184720bc

#1 0x10006b4c

Backtrace stopped.

```

Registers' content
grp00: 0x000000ee 0x7ffd6ad0 0x1800f440 0x00000004
grp04: 0x7ffd6af8 0x0000000a 0xffffffff 0x184720bc
grp08: 0x0002d200 0x00000003 0x00000001 0x1847209c
grp12: 0x10006b4c 0x10020534 0xd6744100 0x00000000
grp16: 0x00000000 0xa0203ff0 0xa028b12c 0xa028b13c
grp20: 0xa028b148 0xa028b168 0xa028b178 0xa028b190
grp24: 0xa028b1a8 0xa028b1b8 0x00000000 0x7ffd6c08
grp28: 0x10006cac 0x7ffd6f92 0x184c1b84 0x7ffd6ae0

```

```

nip:0x184720bc    lr:0x10006b4c    cr:0x38000022    ctr:0x1847209c
msr:0x0002db00   xer:0x00000000   ret:0xffffffffc dsisr:0x08000000
gr3:0x00000003   mq:0x00000000   trap:0x00000c00   dar:0x1833114c

```

# 显示在 powerpc 体系 64 位登录设备上的异常上下文信息。

```

<Sysname> display exception context
Index 1 of 1

```

```

-----
Crashed PID: 172 (routed)
Crash signal: SIGBUS
Crash time: Sat Sep 15 16:53:16 2007
Core file path:
flash:/core/node1_routed_172_7_20070915-165316_1189875196.core
#0  0x00000fff803c66b4
#1  0x0000000010009b94
#2  0x00000fff80401814
Backtrace stopped.

```

```

Registers' content
grp00: 0x00000000000000ee 0x00000ffffd04840
grp02: 0x00000fff80425c28 0x0000000000000004
grp04: 0x00000ffffd048c0 0x000000000000000a
grp06: 0xffffffffffffffff 0x00000fff803c66b4
grp08: 0x000000008002d000 0x0000000000000000
grp10: 0x0000000000000000 0x0000000000000000
grp12: 0x0000000000000000 0x00000fff80a096b0
grp14: 0x000000007b964c00 0x000000007b7d0000
grp16: 0x0000000000000001 0x000000000000000b
grp18: 0x0000000000000031 0x0000000000a205b8
grp20: 0x0000000000a20677 0x0000000000000000
grp22: 0x000000007bb91014 0x0000000000000000
grp24: 0xc000000005ae1c8 0x0000000000000000
grp26: 0xc0000001f00bff20 0xc0000001f00b0000
grp28: 0x00000ffffd04a30 0x000000001001aed8
grp30: 0x00000ffffd04fae 0x00000ffffd04840

```

```

nip:0x00000fff803c66b4    lr:0x0000000010009b94
cr:0x0000000058000482    ctr:0x00000fff803c66ac
msr:0x000000008002d000    xer:0x0000000000000000
ret:0xffffffffffffffffc   dsisr:0x0000000000000000

```

```
gr3:0x0000000000000003      softe:0x0000000000000001
trap:0x00000000000000c00     dar:0x00000fff8059d14c
```

# 显示在 mips 体系 32 位登录设备上的异常上下文信息。

<Sysname> display exception context

Index 1 of 1

-----

Crashed PID: 182 (routed)

Crash signal: SIGBUS

Crash time: Sun Jan 2 08:11:38 2013

Core file path:

flash:/core/node4\_routed\_182\_10\_20130102-081138\_1293955898.core

#0 0x2af2faf4

#1 0x00406d8c

Backtrace stopped.

Registers' content

zero:0x00000000	at:0x1000dc00	v0:0x00000004	v1:0x00000003
a0:0x00000003	a1:0x7fd267e8	a2:0x0000000a	a3:0x00000001
t0:0x00000000	t1:0xcf08fa14	t2:0x80230510	t3:0xfffffffff8
t4:0x69766520	t5:0x00000000	t6:0x63cc6000	t7:0x44617461
s0:0x7fd26f81	s1:0x00401948	s2:0x7fd268f8	s3:0x803e1db0
s4:0x803e1da0	s5:0x803e1d88	s6:0x803e1d70	s7:0x803e1d60
t8:0x00000008	t9:0x2af2fae0	k0:0x00000000	k1:0x00000000
gp:0x2af9a3a0	sp:0x7fd267c0	s8:0x7fd267c0	ra:0x00406d8c
sr:0x0000dc13	lo:0xef9db265	hi:0x0000003f	bad:0x2add2010
cause:0x00800020	pc:0x2af2faf4		

# 显示在 mips 体系 64 位登录设备上的异常上下文信息。

<Sysname> display exception context

Index 1 of 1

-----

Crashed PID: 270 (routed)

Crash signal: SIGBUS

Crash time: Wed Mar 27 12:39:12 2013

Core file path:

flash:/core/node16\_routed\_270\_10\_20130327-123912\_1364387952.core

#0 0x00000005555a3bcb4

#1 0x0000000120006c1c

Backtrace stopped.

Registers' content

zero:0x0000000000000000	at:0x0000000000000014
v0:0x0000000000000004	v1:0x0000000000000003
a0:0x0000000000000003	a1:0x000000ffff899d90
a2:0x000000000000000a	a3:0x0000000000000001
a4:0x00000005555a9b4e0	a5:0x0000000000000000
a6:0xfffffffff8021349c	a7:0x20696e206368616e
t0:0x0000000000000000	t1:0xfffffffff80105068
t2:0xfffffffff80213890	t3:0x0000000000000008
s0:0x00000005555a99c40	s1:0x000000ffff89af5f
s2:0x0000000120007320	s3:0x00000005555a5f470



```

s4:0x000000ffff899f80      s5:0xfffffffff803cc6c0
s6:0xfffffffff803cc6a8      s7:0xfffffffff803cc690
t8:0x0000000000000002      t9:0x0000005555a3bc98
k0:0x0000000000000000      k1:0x0000000000000000
gp:0x0000000120020460      sp:0x000000ffff899d70
s8:0x000000ffff899d80      ra:0x0000000120006c1c
sr:0x000000000400fff3      lo:0xdf3b645a1cac08c9
hi:0x000000000000007f      bad:0x000000555589ba84
cause:0x000000000800020    pc:0x0000005555a3bcb4

```

表1-1 display exception context 命令输出信息描述表

字段	描述
Crashed PID	发生异常的用户态进程ID
Crash signal	导致异常的信号： <ul style="list-style-type: none"> <li>• SIGABRT: 异常终止 (abort)</li> <li>• SIGBUS: 总线错误</li> <li>• SIGFPE: 浮点异常</li> <li>• SIGILL: 程序执行了非法指令，导致异常</li> <li>• SIGQUIT: 终端退出符</li> <li>• SIGSEGV: 无效存储访问</li> <li>• SIGSYS: 无效系统调用</li> <li>• SIGTRAP: 跟踪断点时发生了异常</li> <li>• SIGXCPU: 超过 CPU 限制 (setrlimit)</li> <li>• SIGXFSZ: 超过文件长度限制 (setrlimit)</li> <li>• SIGUNKNOWN: 未知原因</li> </ul>
Crash time	异常发生的时间
Core file path	core文件存放的位置
Backtrace stopped	表示栈信息已经显示完毕
Registers' content	寄存器的内容

## 【相关命令】

- `reset exception context`

### 1.1.2 display exception filepath

`display exception filepath` 命令用来显示 core 文件的保存路径。

## 【命令】

```
display exception filepath [ slot slot-number [ cpu cpu-number ] ]
```

## 【视图】

任意视图

### 【缺省用户角色】

network-admin

### 【参数】

**slot** *slot-number*: 表示设备在 IRF 中的成员编号。不指定该参数时，表示主设备。

**cpu** *cpu-number*: 表示 CPU 的编号。

### 【举例】

# 显示指定 slot 上 core 文件的保存路径。

```
<Sysname> display exception filepath slot 1
The exception filepath on slot 1 is flash:.
```

## 1.1.3 display kernel deadlock

**display kernel deadlock** 命令用来显示内核线程死循环信息。

### 【命令】

```
display kernel deadlock show-number [ offset ] [ verbose ] [ slot slot-number ]
[ cpu cpu-number ]
```

### 【视图】

任意视图

### 【缺省用户角色】

network-admin

### 【参数】

**show-number**: 需要显示的死循环信息的数目，取值范围为 1~20。

**offset**: 需要显示的起始条目距最近条目的偏移，取值范围为 0~19，缺省值为 0。

**verbose**: 表示显示详细信息。不指定该参数时，显示概要信息。

**slot** *slot-number*: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。

**cpu** *cpu-number*: 表示 CPU 的编号。

### 【举例】

# 显示最近一条内核线程死循环的概要信息。

```
<Sysname> display kernel deadlock 1
----- Deadloop record 1 -----
Description           : BUG: soft lockup - CPU#0 stuck for 61! [comsh: 16306]
Recorded at           : 2013-05-01 11:16:00.823018
Occurred at           : 2013-05-01 11:16:00.823018
Instruction address    : 0x4004158c
Thread                : comsh (TID: 16306)
Context               : thread context
Slot                  : 1
Cpu                   : 0
VCPU ID               : 0
Kernel module info    : module name (mrpnc) module address (0xe332a000)
```

# 显示最近一条内核线程死循环的详细信息。

```
<Sysname> display kernel deadlock 1 verbose
```

```
----- Deadloop record 1 -----
```

```
Description      : BUG: soft lockup - CPU#0 stuck for 61! [comsh: 16306]
Recorded at       : 2013-05-01 11:16:00.823018
Occurred at       : 2013-05-01 11:16:00.823018
Instruction address : 0x4004158c
Thread           : comsh (TID: 16306)
Context          : thread context
Slot             : 1
Cpu              : 0
VCPU ID          : 0
Kernel module info : module name (mrpnc) module address (0xe332a000)
```

```
Last 5 thread switches : migration/0 (11:16:00.823018)-->
```

```
swapper (11:16:00.833018)-->
```

```
kthreadd (11:16:00.833518)-->
```

```
swapper (11:16:00.833550)-->
```

```
disk (11:16:00.833560)
```

```
Register content:
```

```
Reg:      r0, Val = 0x00000000 ; Reg:      r1, Val = 0xe2be5ea0 ;
Reg:      r2, Val = 0x00000000 ; Reg:      r3, Val = 0x77777777 ;
Reg:      r4, Val = 0x00000000 ; Reg:      r5, Val = 0x00001492 ;
Reg:      r6, Val = 0x00000000 ; Reg:      r7, Val = 0x0000ffff ;
Reg:      r8, Val = 0x77777777 ; Reg:      r9, Val = 0x00000000 ;
Reg:     r10, Val = 0x00000001 ; Reg:     r11, Val = 0x0000002c ;
Reg:     r12, Val = 0x057d9484 ; Reg:     r13, Val = 0x00000000 ;
Reg:     r14, Val = 0x00000000 ; Reg:     r15, Val = 0x02000000 ;
Reg:     r16, Val = 0xe2be5f00 ; Reg:     r17, Val = 0x00000000 ;
Reg:     r18, Val = 0x00000000 ; Reg:     r19, Val = 0x00000000 ;
Reg:     r20, Val = 0x024c10f8 ; Reg:     r21, Val = 0x057d9244 ;
Reg:     r22, Val = 0x00002000 ; Reg:     r23, Val = 0x0000002c ;
Reg:     r24, Val = 0x00000002 ; Reg:     r25, Val = 0x24000024 ;
Reg:     r26, Val = 0x00000000 ; Reg:     r27, Val = 0x057d9484 ;
Reg:     r28, Val = 0x0000002c ; Reg:     r29, Val = 0x00000000 ;
Reg:     r30, Val = 0x0000002c ; Reg:     r31, Val = 0x00000000 ;
Reg:      cr, Val = 0x84000028 ; Reg:      nip, Val = 0x057d9550 ;
Reg:      xer, Val = 0x00000000 ; Reg:      lr, Val = 0x0186eff0 ;
Reg:      ctr, Val = 0x682f7344 ; Reg:      msr, Val = 0x00784b5c ;
Reg:      trap, Val = 0x0000b030 ; Reg:      dar, Val = 0x77777777 ;
Reg:     dsisr, Val = 0x40000000 ; Reg:     result, Val = 0x00020300 ;
```

```
Dump stack (total 1024 bytes, 16 bytes/line):
```

```
0xe2be5ea0: 02 be 5e c0 24 00 00 24 00 00 00 05 7d 94 84
0xe2be5eb0: 00 00 00 04 00 00 00 00 00 00 00 28 05 8d 34 c4
0xe2be5ec0: 02 be 60 a0 01 86 ef f0 00 00 00 00 00 00 00 00
0xe2be5ed0: 02 04 05 b4 00 00 00 00 00 00 00 00 00 00 00 00
```

```

0xe2be5ee0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5ef0: 95 47 73 35 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5f00: a0 e1 64 21 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5f10: 00 00 00 00 00 00 00 00 00 00 00 00 01 e9 00 00
0xe2be5f20: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5f30: 00 00 00 00 00 00 00 00 02 be 66 c0 02 be 66 d0
0xe2be5f40: 02 be 61 e0 00 00 00 02 00 00 00 00 02 44 b3 a4
0xe2be5f50: 02 be 5f 90 00 00 00 08 02 be 5f e0 00 00 00 08
0xe2be5f60: 02 be 5f 80 00 ac 1b 14 00 00 00 00 00 00 00 00
0xe2be5f70: 05 b4 5f 90 02 be 5f e0 00 00 00 30 02 be 5f e0
0xe2be5f80: 02 be 5f c0 00 ac 1b f4 00 00 00 00 02 45 00 00
0xe2be5f90: 00 03 00 00 00 00 00 00 02 be 5f e0 00 00 00 30
0xe2be5fa0: 02 be 5f c0 00 ac 1b 14 61 f1 2e ae 02 45 00 00
0xe2be5fb0: 02 44 b3 74 02 be 5f d0 00 00 00 30 02 be 5f e0
0xe2be5fc0: 02 be 60 60 01 74 ff f8 00 00 00 00 00 00 08 00
0xe2be5fd0: 02 be 5f f0 00 e8 93 7e 02 be 5f f8 02 be 5f fc
0xe2be5fe0: 00 00 00 00 00 00 00 00 00 00 00 00 02 be 60 18
0xe2be5ff0: 02 be 60 10 00 e9 65 98 00 00 00 58 00 00 2a 4f
0xe2be6000: 02 be 60 10 00 00 00 00 00 00 00 00 02 be 60 68
0xe2be6010: 02 be 60 40 00 e8 c6 a0 00 00 11 17 00 00 00 00
0xe2be6020: 02 be 60 40 00 00 00 00 00 00 00 00 02 be 60 98
0xe2be6030: 02 27 00 00 00 00 00 00 00 00 00 00 02 be 60 68
0xe2be6040: 02 be 60 60 00 00 00 01 00 00 b0 30 02 be 60 98
0xe2be6050: 00 00 00 04 02 21 00 00 00 00 00 00 01 e9 00 00
0xe2be6060: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be6070: 00 00 00 00 00 00 00 00 02 be 66 c0 02 be 66 d0
0xe2be6080: 02 be 61 e0 00 00 00 02 00 00 00 00 02 be 61 70
0xe2be6090: 00 00 00 00 02 21 00 00 05 8d 34 c4 05 7d 92 44

```

#### Call trace:

```

Function Address = 0x8012a4b4
Function Address = 0x8017989c
Function Address = 0x80179b30
Function Address = 0x80127438
Function Address = 0x8012d734
Function Address = 0x80100a00
Function Address = 0xe0071004
Function Address = 0x8016ce0c
Function Address = 0x801223a0

```

#### Instruction dump:

```

41a2fe9c 812300ec 800200ec 7f890000 409efe8c 80010014 540b07b9 40a2fe80
4bffffe6c 80780290 7f64db78 4804ea35 <807f002c> 38800000 38a00080 3863000c

```

表1-2 display kernel deadloop 命令显示信息描述表

字段	描述
Description	发生死循环的内核线程的描述信息，包括死循环内核线程所在的CPU的编号、内核线程连续运行的时间、内核线程的名称和编号

字段	描述
Recorded at	内核线程死循环被记录的时间点，精确到微秒
Occurred at	内核线程发生死循环的时间，精确到微秒
Instruction address	内核线程被检测到发生死循环时对应的指令信息
Thread	发生死循环的内核线程的名称和编号
Context	内核线程被检测到发生死循环时所在的上下文环境
Cpu	运行该内核线程的CPU的编号
VCPU ID	运行该内核线程的CPU核的编号
Kernel module info	内核线程被检测到发生死循环时，系统中已加载的内核模块信息。包括： <ul style="list-style-type: none"> <li><code>module name</code> 表示内核模块名称</li> <li><code>module address</code> 内核模块加载的内存地址</li> </ul>
Last 5 thread switches	内核线程被检测到发生死循环时，记录死循环发生的CPU上、最近五次的内核线程切换轨迹。包括内核线程的名称和内核线程切换时间点，时间精确到微秒
Register content	内核线程被检测到发生死循环时现场的寄存器信息。 <code>Reg</code> 表示寄存器名称， <code>Val</code> 表示寄存器中保存的值
Dump stack	内核线程被检测到发生死循环时现场的堆栈信息
Call trace	内核线程被检测到发生死循环时现场的函数调用栈信息，即每级调用函数的指令地址
Instruction dump	内核线程被检测到发生死循环时对应的指令码。非法指令用 <code>ffffff</code> 表示
No information to display	表示系统中没有内核线程死循环记录

### 【相关命令】

- `reset kernel deadlock`

#### 1.1.4 display kernel deadlock configuration

`display kernel deadlock configuration` 命令用来显示内核线程死循环监控参数配置。

### 【命令】

`display kernel deadlock configuration [ slot slot-number [ cpu cpu-number ] ]`

### 【视图】

任意视图

### 【缺省用户角色】

network-admin

### 【参数】

`slot slot-number`: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。

`cpu cpu-number`: 表示 CPU 的编号。

【举例】

```
# 显示内核线程死循环监控参数配置。
<Sysname> display kernel deadlock configuration
Thread dead loop detection: Enabled
Dead loop timer (in seconds): 20
Dead loop core list: 0-1
Dead loop action: Record-only
Threads excluded from monitoring: 1
    TID:      15    Name: co0
```

表1-3 display kernel deadlock configuration 命令显示信息描述表

字段	描述
Thread dead loop detection: Enabled	内核线程死循环检测功能处于开启状态
Thread dead loop detection: Disabled	内核线程死循环检测功能处于关闭状态
Dead loop timer (in seconds): <i>n</i>	内核线程死循环判定周期（单位为秒），即内核线程连续运行时间大于 <i>n</i> 秒时，则判定为死循环
Dead loop core list	表示系统需要检测是否发生内核死循环的CPU核的编号
Dead loop action:	内核线程被判定为死循环后系统执行的动作，取值为： <ul style="list-style-type: none"><li>• Reboot: 记录日志并重启</li><li>• Record-only: 只记录日志，继续运行</li></ul>
Threads excluded from monitoring	不进行死循环检测的内核线程列表，配置 <b>monitor kernel deadlock exclude-thread</b> 命令后才会显示该信息
Name	不进行死循环检测的内核线程的名称
TID	不进行死循环检测的内核线程的编号
No thread is excluded from monitoring	对所有内核线程都进行死循环检查

1.1.5 display kernel exception

**display kernel exception** 命令用来显示内核线程的异常信息。

【命令】

```
display kernel exception show-number [ offset ] [ verbose ] [ slot slot-number ] [ cpu cpu-number ]
```

【视图】

任意视图

【缺省用户角色】

network-admin

## 【参数】

**show-number**: 需要显示的异常信息的数目，取值范围为 1~20。

**offset**: 开始显示的条目距最近条目的偏移，取值范围为 0~19，缺省值为 0。

**verbose**: 显示详细信息。不指定该参数时，显示概要信息。

**slot slot-number**: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。

**cpu cpu-number**: 表示 CPU 的编号。

## 【使用指导】

当内核线程在运行过程中发生异常时，系统会自动记录异常信息，以便设备维护人员定位问题。

## 【举例】

# 显示最近一条内核线程异常的概要信息。

```
<Sysname> display kernel exception 1
----- Exception record 1 -----
Description          : Oops[#0]
Recorded at          : 2013-05-01 11:16:00.823018
Occurred at          : 2013-05-01 11:16:00.823018
Instruction address   : 0x4004158c
Thread               : comsh (TID: 16306)
Context              : thread context
Slot                 : 1
Cpu                  : 0
VCPU ID              : 0
Kernel module info   : module name (mrpnc) module address (0xe332a000)
                      : module name (disk) module address (0xe00bd000)
```

# 显示最近一条内核线程异常的详细信息。

```
<Sysname> display kernel exception 1 verbose
----- Exception record 1 -----
Description          : Oops[#0]
Recorded at          : 2013-05-01 11:16:00.823018
Occurred at          : 2013-05-01 11:16:00.823018
Instruction address   : 0x4004158c
Thread               : comsh (TID: 16306)
Context              : thread context
Slot                 : 1
Cpu                  : 0
VCPU ID              : 0
Kernel module info   : module name (mrpnc) module address (0xe332a000)
                      : module name (l2500) module address (0xe00bd000)
```

```
Last 5 thread switches : migration/0 (11:16:00.823018)-->
                        swapper (11:16:00.833018)-->
                        kthreadd (11:16:00.833518)-->
                        swapper (11:16:00.833550)-->
                        disk (11:16:00.833560)
```

Register content:

```

Reg:      r0, Val = 0x00000000 ; Reg:      r1, Val = 0xe2be5ea0 ;
Reg:      r2, Val = 0x00000000 ; Reg:      r3, Val = 0x77777777 ;
Reg:      r4, Val = 0x00000000 ; Reg:      r5, Val = 0x00001492 ;
Reg:      r6, Val = 0x00000000 ; Reg:      r7, Val = 0x0000ffff ;
Reg:      r8, Val = 0x77777777 ; Reg:      r9, Val = 0x00000000 ;
Reg:      r10, Val = 0x00000001 ; Reg:     r11, Val = 0x0000002c ;
Reg:     r12, Val = 0x057d9484 ; Reg:     r13, Val = 0x00000000 ;
Reg:     r14, Val = 0x00000000 ; Reg:     r15, Val = 0x02000000 ;
Reg:     r16, Val = 0xe2be5f00 ; Reg:     r17, Val = 0x00000000 ;
Reg:     r18, Val = 0x00000000 ; Reg:     r19, Val = 0x00000000 ;
Reg:     r20, Val = 0x024c10f8 ; Reg:     r21, Val = 0x057d9244 ;
Reg:     r22, Val = 0x00002000 ; Reg:     r23, Val = 0x0000002c ;
Reg:     r24, Val = 0x00000002 ; Reg:     r25, Val = 0x24000024 ;
Reg:     r26, Val = 0x00000000 ; Reg:     r27, Val = 0x057d9484 ;
Reg:     r28, Val = 0x0000002c ; Reg:     r29, Val = 0x00000000 ;
Reg:     r30, Val = 0x0000002c ; Reg:     r31, Val = 0x00000000 ;
Reg:      cr, Val = 0x84000028 ; Reg:     nip, Val = 0x057d9550 ;
Reg:     xer, Val = 0x00000000 ; Reg:      lr, Val = 0x0186eff0 ;
Reg:      ctr, Val = 0x682f7344 ; Reg:     msr, Val = 0x00784b5c ;
Reg:     trap, Val = 0x0000b030 ; Reg:     dar, Val = 0x77777777 ;
Reg:     dsisr, Val = 0x40000000 ; Reg:    result, Val = 0x00020300 ;

```

Dump stack (total 1024 bytes, 16 bytes/line):

```

0xe2be5ea0: 02 be 5e c0 24 00 00 24 00 00 00 05 7d 94 84
0xe2be5eb0: 00 00 00 04 00 00 00 00 00 00 00 28 05 8d 34 c4
0xe2be5ec0: 02 be 60 a0 01 86 ef f0 00 00 00 00 00 00 00
0xe2be5ed0: 02 04 05 b4 00 00 00 00 00 00 00 00 00 00 00
0xe2be5ee0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5ef0: 95 47 73 35 00 00 00 00 00 00 00 00 00 00 00
0xe2be5f00: a0 e1 64 21 00 00 00 00 00 00 00 00 00 00 00
0xe2be5f10: 00 00 00 00 00 00 00 00 00 00 00 00 01 e9 00 00
0xe2be5f20: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5f30: 00 00 00 00 00 00 00 00 02 be 66 c0 02 be 66 d0
0xe2be5f40: 02 be 61 e0 00 00 00 02 00 00 00 00 02 44 b3 a4
0xe2be5f50: 02 be 5f 90 00 00 00 08 02 be 5f e0 00 00 00 08
0xe2be5f60: 02 be 5f 80 00 ac 1b 14 00 00 00 00 00 00 00 00
0xe2be5f70: 05 b4 5f 90 02 be 5f e0 00 00 00 30 02 be 5f e0
0xe2be5f80: 02 be 5f c0 00 ac 1b f4 00 00 00 00 02 45 00 00
0xe2be5f90: 00 03 00 00 00 00 00 00 02 be 5f e0 00 00 00 30
0xe2be5fa0: 02 be 5f c0 00 ac 1b 14 61 f1 2e ae 02 45 00 00
0xe2be5fb0: 02 44 b3 74 02 be 5f d0 00 00 00 30 02 be 5f e0
0xe2be5fc0: 02 be 60 60 01 74 ff f8 00 00 00 00 00 00 08 00
0xe2be5fd0: 02 be 5f f0 00 e8 93 7e 02 be 5f f8 02 be 5f fc
0xe2be5fe0: 00 00 00 00 00 00 00 00 00 00 00 00 02 be 60 18
0xe2be5ff0: 02 be 60 10 00 e9 65 98 00 00 00 58 00 00 2a 4f
0xe2be6000: 02 be 60 10 00 00 00 00 00 00 00 00 02 be 60 68
0xe2be6010: 02 be 60 40 00 e8 c6 a0 00 00 11 17 00 00 00 00
0xe2be6020: 02 be 60 40 00 00 00 00 00 00 00 00 02 be 60 98

```



```
0xe2be6030: 02 27 00 00 00 00 00 00 00 00 00 00 02 be 60 68
0xe2be6040: 02 be 60 60 00 00 00 01 00 00 b0 30 02 be 60 98
0xe2be6050: 00 00 00 04 02 21 00 00 00 00 00 00 01 e9 00 00
0xe2be6060: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be6070: 00 00 00 00 00 00 00 00 02 be 66 c0 02 be 66 d0
0xe2be6080: 02 be 61 e0 00 00 00 02 00 00 00 00 02 be 61 70
0xe2be6090: 00 00 00 00 02 21 00 00 05 8d 34 c4 05 7d 92 44
```

Call trace:

```
Function Address = 0x8012a4b4
Function Address = 0x8017989c
Function Address = 0x80179b30
Function Address = 0x80127438
Function Address = 0x8012d734
Function Address = 0x80100a00
Function Address = 0xe0071004
Function Address = 0x8016ce0c
Function Address = 0x801223a0
```

Instruction dump:

```
41a2fe9c 812300ec 800200ec 7f890000 409efe8c 80010014 540b07b9 40a2fe80
4bffffe6c 80780290 7f64db78 4804ea35 <807f002c> 38800000 38a00080 3863000c
```

本命令显示信息的详细描述请参见 [表 1-2](#)。

## 【相关命令】

- **reset kernel exception**

## 1.1.6 display kernel reboot

**display kernel reboot** 命令用来显示成员设备的重启信息。

## 【命令】

```
display kernel reboot show-number [ offset ] [ verbose ] [ slot slot-number ]
[ cpu cpu-number ]
```

## 【视图】

任意视图

## 【缺省用户角色】

network-admin

## 【参数】

**show-number**: 需要显示的重启信息的数目，取值范围为 1~20。

**offset**: 需要显示的起始条目距最近条目的偏移，取值范围为 0~19，缺省值为 0。

**verbose**: 表示显示详细信息。不指定该参数时，显示概要信息。

**slot slot-number**: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。成员设备的重启信息会记录在 IRF 当前主设备的内存中，主设备掉电后会删除该信息。

**cpu cpu-number**: 表示 CPU 的编号。

## 【举例】

# 显示最近一条重启概要信息。

```
<Sysname> display kernel reboot 1
----- Reboot record 1 -----
Recorded at      : 2013-05-01 11:16:00.823018
Occurred at      : 2013-05-01 11:16:00.823018
Reason           : 0x31
Thread           : comsh (TID: 16306)
Context          : thread context
Slot             : 1
Target Slot      : 0
Cpu              : 0
VCPU ID          : 0
Kernel module info : module name (mrpnc) module address (0xe332a000)
                   : module name (12500) module address (0xe00bd000)
```

# 显示最近一条重启详细信息。

```
<Sysname> display kernel reboot 1 verbose
----- Reboot record 1 -----
Recorded at      : 2013-05-01 11:16:00.823018
Occurred at      : 2013-05-01 11:16:00.823018
Reason           : 0x31
Thread           : comsh (TID: 16306)
Context          : thread context
Slot             : 1
Target Slot      : 0
Cpu              : 0
VCPU ID          : 0
Kernel module info : module name (mrpnc) module address (0xe332a000)
                   : module name (12500) module address (0xe00bd000)
```

```
Last 5 thread switches : migration/0 (11:16:00.823018)-->
                        swapper (11:16:00.833018)-->
                        kthreadd (11:16:00.833518)-->
                        swapper (11:16:00.833550)-->
                        disk (11:16:00.833560)
```

Dump stack (total 1024 bytes, 16 bytes/line):

```
0xe2be5ea0: 02 be 5e c0 24 00 00 24 00 00 00 05 7d 94 84
0xe2be5eb0: 00 00 00 04 00 00 00 00 00 00 00 28 05 8d 34 c4
0xe2be5ec0: 02 be 60 a0 01 86 ef f0 00 00 00 00 00 00 00 00
0xe2be5ed0: 02 04 05 b4 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5ee0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5ef0: 95 47 73 35 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5f00: a0 e1 64 21 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5f10: 00 00 00 00 00 00 00 00 00 00 00 00 01 e9 00 00
0xe2be5f20: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5f30: 00 00 00 00 00 00 00 00 02 be 66 c0 02 be 66 d0
```

```

0xe2be5f40: 02 be 61 e0 00 00 00 02 00 00 00 00 02 44 b3 a4
0xe2be5f50: 02 be 5f 90 00 00 00 08 02 be 5f e0 00 00 00 08
0xe2be5f60: 02 be 5f 80 00 ac 1b 14 00 00 00 00 00 00 00 00
0xe2be5f70: 05 b4 5f 90 02 be 5f e0 00 00 00 30 02 be 5f e0
0xe2be5f80: 02 be 5f c0 00 ac 1b f4 00 00 00 00 02 45 00 00
0xe2be5f90: 00 03 00 00 00 00 00 00 02 be 5f e0 00 00 00 30
0xe2be5fa0: 02 be 5f c0 00 ac 1b 14 61 f1 2e ae 02 45 00 00
0xe2be5fb0: 02 44 b3 74 02 be 5f d0 00 00 00 30 02 be 5f e0
0xe2be5fc0: 02 be 60 60 01 74 ff f8 00 00 00 00 00 00 08 00
0xe2be5fd0: 02 be 5f f0 00 e8 93 7e 02 be 5f f8 02 be 5f fc
0xe2be5fe0: 00 00 00 00 00 00 00 00 00 00 00 00 02 be 60 18
0xe2be5ff0: 02 be 60 10 00 e9 65 98 00 00 00 58 00 00 2a 4f
0xe2be6000: 02 be 60 10 00 00 00 00 00 00 00 00 02 be 60 68
0xe2be6010: 02 be 60 40 00 e8 c6 a0 00 00 11 17 00 00 00 00
0xe2be6020: 02 be 60 40 00 00 00 00 00 00 00 00 02 be 60 98
0xe2be6030: 02 27 00 00 00 00 00 00 00 00 00 00 02 be 60 68
0xe2be6040: 02 be 60 60 00 00 00 01 00 00 b0 30 02 be 60 98
0xe2be6050: 00 00 00 04 02 21 00 00 00 00 00 00 01 e9 00 00
0xe2be6060: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be6070: 00 00 00 00 00 00 00 00 02 be 66 c0 02 be 66 d0
0xe2be6080: 02 be 61 e0 00 00 00 02 00 00 00 00 02 be 61 70
0xe2be6090: 00 00 00 00 02 21 00 00 05 8d 34 c4 05 7d 92 44

```

Call trace:

```

Function Address = 0x8012a4b4
Function Address = 0x8017989c
Function Address = 0x80179b30
Function Address = 0x80127438
Function Address = 0x8012d734
Function Address = 0x80100a00
Function Address = 0xe0071004
Function Address = 0x8016ce0c
Function Address = 0x801223a0

```

表1-4 display kernel reboot 命令显示信息描述表

字段	描述
Recorded at	重启被记录的时间点，精确到微秒
Occurred at	重启的时间，精确到微秒
Reason	重启的原因
Thread	重启时运行的内核线程的名称和编号
Context	重启时所在的上下文环境
Slot	触发重启事件的Slot的编号
Target Slot	实际发生重启的Slot的编号
Cpu	触发重启事件的CPU的编号

字段	描述
VCPU ID	触发重启事件的CPU核的编号
Kernel module info	重启发生时，系统中已加载的内核模块信息。包括内核模块名和内核模块加载的内存地址
Last 5 thread switches	重启时，记录重启的CPU上、最近五次的内核线程切换轨迹。包括内核线程的名称和内核线程切换时间点，时间精确到微秒
Dump stack	重启时，运行线程的堆栈信息
Call trace	重启时，运行线程的函数调用栈信息
No information to display	表示系统中没有重启记录

### 【相关命令】

- `reset kernel reboot`

### 1.1.7 display kernel starvation

`display kernel starvation` 命令用来显示内核线程饿死信息。

### 【命令】

```
display kernel starvation show-number [ offset ] [ verbose ] [ slot
slot-number [ cpu cpu-number ] ]
```

### 【视图】

任意视图

### 【缺省用户角色】

network-admin

### 【参数】

**show-number**: 需要显示的饿死信息的数目，取值范围为 1~20。

**offset**: 需要显示的起始条目距最近条目的偏移，取值范围为 0~19，缺省值为 0。

**verbose**: 表示显示详细信息。不指定该参数时，显示概要信息。

**slot slot-number**: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。

**cpu cpu-number**: 表示 CPU 的编号。

### 【举例】

# 显示最近一条内核线程饿死的概要信息。

```
<Sysname> display kernel starvation 1
----- Starvation record 1 -----
Description      : INFO: task comsh: 16306 blocked for more than 10 seconds.
Recorded at      : 2013-05-01 11:16:00.823018
Occurred at      : 2013-05-01 11:16:00.823018
Instruction address : 0x4004158c
Thread           : comsh (TID: 16306)
Context          : thread context
```

```

Slot                : 1
Cpu                 : 0
VCPU ID             : 0
Kernel module info  : module name (mrpnc) module address (0xe332a000)
                     : module name (l2500) module address (0xe00bd000)

```

# 显示最近一条内核线程饿死的详细信息。

```
<Sysname> display kernel starvation 1 verbose
```

```
----- Starvation record 1 -----
```

```

Description          : INFO: task comsh: 16306 blocked for more than 10 seconds.
Recorded at          : 2013-05-01 11:16:00.823018
Occurred at          : 2013-05-01 11:16:00.823018
Instruction address   : 0x4004158c
Thread               : comsh (TID: 16306)
Context              : thread context
Slot                 : 1
Cpu                  : 0
VCPU ID              : 0
Kernel module info    : module name (mrpnc) module address (0xe332a000)
                       : module name (l2500) module address (0xe00bd000)

```

```

Last 5 thread switches : migration/0 (11:16:00.823018)-->
                        swapper (11:16:00.833018)-->
                        kthreadd (11:16:00.833518)-->
                        swapper (11:16:00.833550)-->
                        disk (11:16:00.833560)

```

Register content:

```

Reg:      r0, Val = 0x00000000 ; Reg:      r1, Val = 0xe2be5ea0 ;
Reg:      r2, Val = 0x00000000 ; Reg:      r3, Val = 0x77777777 ;
Reg:      r4, Val = 0x00000000 ; Reg:      r5, Val = 0x00001492 ;
Reg:      r6, Val = 0x00000000 ; Reg:      r7, Val = 0x0000ffff ;
Reg:      r8, Val = 0x77777777 ; Reg:      r9, Val = 0x00000000 ;
Reg:     r10, Val = 0x00000001 ; Reg:     r11, Val = 0x0000002c ;
Reg:     r12, Val = 0x057d9484 ; Reg:     r13, Val = 0x00000000 ;
Reg:     r14, Val = 0x00000000 ; Reg:     r15, Val = 0x02000000 ;
Reg:     r16, Val = 0xe2be5f00 ; Reg:     r17, Val = 0x00000000 ;
Reg:     r18, Val = 0x00000000 ; Reg:     r19, Val = 0x00000000 ;
Reg:     r20, Val = 0x024c10f8 ; Reg:     r21, Val = 0x057d9244 ;
Reg:     r22, Val = 0x00002000 ; Reg:     r23, Val = 0x0000002c ;
Reg:     r24, Val = 0x00000002 ; Reg:     r25, Val = 0x24000024 ;
Reg:     r26, Val = 0x00000000 ; Reg:     r27, Val = 0x057d9484 ;
Reg:     r28, Val = 0x0000002c ; Reg:     r29, Val = 0x00000000 ;
Reg:     r30, Val = 0x0000002c ; Reg:     r31, Val = 0x00000000 ;
Reg:      cr, Val = 0x84000028 ; Reg:      nip, Val = 0x057d9550 ;
Reg:      xer, Val = 0x00000000 ; Reg:      lr, Val = 0x0186eff0 ;
Reg:      ctr, Val = 0x682f7344 ; Reg:      msr, Val = 0x00784b5c ;
Reg:      trap, Val = 0x0000b030 ; Reg:      dar, Val = 0x77777777 ;
Reg:     dsisr, Val = 0x40000000 ; Reg:     result, Val = 0x00020300 ;

```

Dump stack (total 1024 bytes, 16 bytes/line):

```
0xe2be5ea0: 02 be 5e c0 24 00 00 24 00 00 00 00 05 7d 94 84
0xe2be5eb0: 00 00 00 04 00 00 00 00 00 00 00 00 28 05 8d 34 c4
0xe2be5ec0: 02 be 60 a0 01 86 ef f0 00 00 00 00 00 00 00 00
0xe2be5ed0: 02 04 05 b4 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5ee0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5ef0: 95 47 73 35 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5f00: a0 e1 64 21 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5f10: 00 00 00 00 00 00 00 00 00 00 00 00 01 e9 00 00
0xe2be5f20: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be5f30: 00 00 00 00 00 00 00 00 02 be 66 c0 02 be 66 d0
0xe2be5f40: 02 be 61 e0 00 00 00 02 00 00 00 00 02 44 b3 a4
0xe2be5f50: 02 be 5f 90 00 00 00 08 02 be 5f e0 00 00 00 08
0xe2be5f60: 02 be 5f 80 00 ac 1b 14 00 00 00 00 00 00 00 00
0xe2be5f70: 05 b4 5f 90 02 be 5f e0 00 00 00 30 02 be 5f e0
0xe2be5f80: 02 be 5f c0 00 ac 1b f4 00 00 00 00 02 45 00 00
0xe2be5f90: 00 03 00 00 00 00 00 00 02 be 5f e0 00 00 00 30
0xe2be5fa0: 02 be 5f c0 00 ac 1b 14 61 f1 2e ae 02 45 00 00
0xe2be5fb0: 02 44 b3 74 02 be 5f d0 00 00 00 30 02 be 5f e0
0xe2be5fc0: 02 be 60 60 01 74 ff f8 00 00 00 00 00 00 08 00
0xe2be5fd0: 02 be 5f f0 00 e8 93 7e 02 be 5f f8 02 be 5f fc
0xe2be5fe0: 00 00 00 00 00 00 00 00 00 00 00 00 02 be 60 18
0xe2be5ff0: 02 be 60 10 00 e9 65 98 00 00 00 58 00 00 2a 4f
0xe2be6000: 02 be 60 10 00 00 00 00 00 00 00 00 02 be 60 68
0xe2be6010: 02 be 60 40 00 e8 c6 a0 00 00 11 17 00 00 00 00
0xe2be6020: 02 be 60 40 00 00 00 00 00 00 00 00 02 be 60 98
0xe2be6030: 02 27 00 00 00 00 00 00 00 00 00 00 02 be 60 68
0xe2be6040: 02 be 60 60 00 00 00 01 00 00 b0 30 02 be 60 98
0xe2be6050: 00 00 00 04 02 21 00 00 00 00 00 00 01 e9 00 00
0xe2be6060: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0xe2be6070: 00 00 00 00 00 00 00 00 02 be 66 c0 02 be 66 d0
0xe2be6080: 02 be 61 e0 00 00 00 02 00 00 00 00 02 be 61 70
0xe2be6090: 00 00 00 00 02 21 00 00 05 8d 34 c4 05 7d 92 44
```

Call trace:

```
Function Address = 0x8012a4b4
Function Address = 0x8017989c
Function Address = 0x80179b30
Function Address = 0x80127438
Function Address = 0x8012d734
Function Address = 0x80100a00
Function Address = 0xe0071004
Function Address = 0x8016ce0c
Function Address = 0x801223a0
```

Instruction dump:

```
41a2fe9c 812300ec 800200ec 7f890000 409efe8c 80010014 540b07b9 40a2fe80
```

4bffffe6c 80780290 7f64db78 4804ea35 <807f002c> 38800000 38a00080 3863000c  
本命令显示信息的详细描述请参见 [表 1-2](#)。

【相关命令】

- `reset kernel starvation`

1.1.8 display kernel starvation configuration

`display kernel starvation configuration` 命令用来显示内核线程的饿死监控参数的配置。

【命令】

`display kernel starvation configuration [ slot slot-number [ cpu  
cpu-number ] ]`

【视图】

任意视图

【缺省用户角色】

network-admin

【参数】

`slot slot-number`: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。

`cpu cpu-number`: 表示 CPU 的编号。

【举例】

```
# 显示内核线程饿死监控参数配置。
<Sysname> display kernel starvation configuration
Thread starvation detection: Disabled
Starvation timer (in seconds): 10
Threads excluded from monitoring: 1
  TID:    123    Name: co0
```

表1-5 display kernel starvation configuration 命令显示信息描述表

字段	描述
Thread starvation detection: Enabled	内核线程饿死检测功能处于开启状态
Thread starvation detection: Disabled	内核线程饿死检测功能处于关闭状态
Starvation timer (in seconds): <i>n</i>	内核线程饿死判定周期（单位为秒）。即如果内核线程在 <i>n</i> 秒内一直不能运行，则判定为饿死
Threads excluded from monitoring	不进行饿死检测的内核线程列表
Name	不进行饿死检测的内核线程的名称
TID	不进行饿死检测的内核线程的编号

【相关命令】

- `monitor kernel starvation enable`

- **monitor kernel starvation exclude-thread**
- **monitor kernel starvation time**

### 1.1.9 display process

**display process** 命令用来显示进程的状态信息。

#### 【命令】

```
display process [ all | job job-id | name process-name ] [ slot slot-number ]
[ cpu cpu-number ] ]
```

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

**all**: 显示所有进程的状态信息。指定 **all** 参数和不指定任何可选参数时，命令行的执行效果相同。

**job** *job-id*: 任务编号，用于唯一标识一个进程，该编号不会随着进程的重启而改变，取值范围为 1~2147483647。

**name** *process-name*: 进程名称，为 1~15 个字符的字符串，不区分大小写，不能包含问号和空格。

**slot** *slot-number*: 表示设备在 IRF 中的成员编号。不指定该参数时，表示主设备。

**cpu** *cpu-number*: 表示 CPU 的编号。

#### 【举例】

# 显示进程 **scmd** 的状态信息。

```
<Sysname> display process name scmd
      Job ID: 1
      PID: 1
      Parent JID: 0
      Parent PID: 0
      Executable path: /sbin/scmd
      Instance: 0
      Respawn: OFF
      Respawn count: 1
      Max. spawns per minute: 0
      Last started: Wed Jun  1 14:45:46 2013
      Process state: sleeping
      Max. core: 0
      ARGS: -
TID   LAST_CPU   Stack   PRI   State   HH:MM:SS:MSEC   Name
  1       0       OK     120    S       0:0:5:220      scmd
```



表1-6 display process name 命令显示信息描述表

字段	描述
Job ID	任务编号，用于唯一标识一个进程，该编号不会随着进程的重启而改变
PID	进程编号，用于标识一个进程，但该编号可能会随着进程的重启而改变
Parent JID	父进程的任务编号
Parent PID	父进程的进程编号
Executable path	进程执行路径（内核线程执行路径显示为“-”）
Instance	进程的实例号（一个进程根据需要在软件实现时决定了它是否会运行多个实例）
Respawn	运行出错时，该进程是否会自动重启： <ul style="list-style-type: none"> <li>ON 表示自动重启</li> <li>OFF 表示不自动重启</li> </ul>
Respawn count	进程重启的次数（初始值为1）
Max. spawns per minute	进程一分钟内允许异常重启的最大次数（如果进程在一分钟内异常重启次数超过该值，则系统会自动关闭该进程）
Last started	进程最近一次启动的日期和时间
Process state	进程状态，可能的取值为： <ul style="list-style-type: none"> <li>running: 运行状态或正在队列中等待调度</li> <li>sleeping: 可中断睡眠状态</li> <li>traced or stopped: 暂停状态</li> <li>uninterruptible sleep: 不可中断睡眠状态</li> <li>zombie: 僵死状态（僵死状态指的是进程已经退出，但是仍然占用部分资源的状态）</li> </ul>
Max. core	进程最多可以生成的core文件的数量，如果为0表示不生成core文件（进程异常重启一次，会产生一个core文件。如果生成的core文件的数目达到最大值，则不再生成core文件。软件开发和维护人员能够根据core文件的内容来定位异常的原因和异常的位置）
ARGS	进程启动时携带的参数。如果进程不带参数，显示为“-”
TID	线程编号
LAST_CPU	进程最近一次被调度时，所在的CPU
Stack	堆栈大小
PRI	线程优先级
State	线程状态，可能的取值为： <ul style="list-style-type: none"> <li>R: running, 运行状态或正在队列中等待调度</li> <li>S: sleeping, 可中断睡眠状态</li> <li>T: traced or stopped, 暂停状态</li> <li>D: uninterruptible sleep, 不可中断睡眠状态</li> <li>Z: zombie, 僵死状态</li> </ul>
HH:MM:SS:MSEC	进程最近一次启动后的运行时间

字段	描述
Name	进程名称

# 显示所有进程的状态信息。

```
<Sysname> display process all
  JID   PID  %CPU %MEM  STAT PRI  THIRD TTY  HH:MM:SS COMMAND
    1     1   0.0  0.0   S  120   N    -   00:00:04 scmd
    2     2   0.0  0.0   S  115   N    -   00:00:00 [kthreadd]
    3     3   0.0  0.0   S   99   N    -   00:00:00 [migration/0]
    4     4   0.0  0.0   S  115   N    -   00:00:05 [ksoftirqd/0]
    5     5   0.0  0.0   S   99   N    -   00:00:00 [watchdog/0]
    6     6   0.0  0.0   S  115   N    -   00:00:00 [events/0]
    7     7   0.0  0.0   S  115   N    -   00:00:00 [khelper]
    8     8   0.0  0.0   S  115   N    -   00:00:00 [kblockd/0]
    9     9   0.0  0.0   S  115   N    -   00:00:00 [ata/0]
   10    10   0.0  0.0   S  115   N    -   00:00:00 [ata_aux]
   11    11   0.0  0.0   S  115   N    -   00:00:00 [kseriod]
   12    12   0.0  0.0   S  120   N    -   00:00:00 [vzmond]
   13    13   0.0  0.0   S  120   N    -   00:00:00 [pdflush]
   14    14   0.0  0.0   S  120   N    -   00:00:00 [pdflush]
   15    15   0.0  0.0   S  115   N    -   00:00:00 [kswapd0]
   16    16   0.0  0.0   S  115   N    -   00:00:00 [aio/0]
   17    17   0.0  0.0   S  115   N    -   00:00:00 [scsi_eh_0]
   18    18   0.0  0.0   S  115   N    -   00:00:00 [scsi_eh_1]
   19    19   0.0  0.0   S  115   N    -   00:00:00 [scsi_eh_2]
   35    35   0.0  0.0   D  100   N    -   00:00:00 [lipc_topology]

---- More ----
```

表1-7 display process all 命令显示信息描述

字段	描述
JID	任务编号，用于唯一标识一个进程，该编号不会随着进程的重启而改变
PID	进程编号
%CPU	CPU使用率（用百分比表示）
%MEM	内存使用率（用百分比表示）
STAT	进程状态，可能的取值为： <ul style="list-style-type: none"> <li>• R: running，运行状态或处于运行队列</li> <li>• S: sleeping，可中断睡眠状态</li> <li>• T: traced or stopped，暂停状态</li> <li>• D: uninterruptible sleep，不可中断睡眠状态</li> <li>• Z: zombie，僵死状态</li> </ul>
PRI	进程优先级（优先级在进程调度时发挥作用，优先级高的会优先得到调度）

字段	描述
THIRD	（暂不支持）是否为第三方程序： <ul style="list-style-type: none"> <li>Y: 是第三方程序</li> <li>N: 不是第三方程序</li> </ul>
TTY	进程使用的终端
HH:MM:SS	进程最近一次启动后的运行时间。当进程的持续运行时间大于或等于100小时时，该列仅显示小时数，不再显示分和秒
COMMAND	进程名称以及进程运行的参数（如果进程名称带有“[]”标记，则表示内核线程）

### 1.1.10 display process cpu

**display process cpu** 命令用来显示所有进程的 CPU 使用率信息。

#### 【命令】

**display process cpu** [ **slot** *slot-number* [ **cpu** *cpu-number* ] ]

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

**slot** *slot-number*: 表示设备在 IRF 中的成员编号。不指定该参数时，表示主设备。

**cpu** *cpu-number*: 表示 CPU 的编号。

#### 【举例】

# 显示所有进程 CPU 使用率信息。

```
<Sysname> display process cpu
CPU utilization in 5 secs: 16.8%; 1 min: 4.7%; 5 mins: 4.7%
  JID      5Sec      1Min      5Min      Name
    1      0.0%      0.0%      0.0%      scmd
    2      0.0%      0.0%      0.0%      [kthreadd]
    3      0.1%      0.0%      0.0%      [ksoftirqd/0]
```

其它显示信息略……。

表1-8 display process cpu 命令显示信息描述表

字段	描述
CPU utilization in 5 secs: 16.8%; 1 min: 4.7%; 5 mins: 4.7%	系统最近5秒CPU使用率；最近1分钟CPU使用率；最近5分钟CPU使用率
JID	任务编号（用于唯一标识一个进程，该编号不会随着进程的重启而改变）
5Sec	最近5秒钟内进程的CPU使用率

字段	描述
1Min	最近1分钟内进程的CPU使用率
5Min	最近5分钟内进程的CPU使用率
Name	进程名称（如果进程名称带有“[]”标记，则表示该进程为内核线程）

### 1.1.11 display process log

**display process log** 命令用来显示所有用户态进程的日志信息。

#### 【命令】

**display process log** [ **slot** *slot-number* [ **cpu** *cpu-number* ] ]

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

**slot** *slot-number*: 表示设备在 IRF 中的成员编号。不指定该参数时，表示主设备。

**cpu** *cpu-number*: 表示 CPU 的编号。

#### 【举例】

# 显示所有用户态进程的日志信息。

```
<Sysname> display process log
```

Process	JobID	PID	Abort	Core	Exit	Kill	StartTime	EndTime
knotify	92	92	N	N	0	36	12-17 07:10:27	12-17 07:10:27
knotify	93	93	N	N	0	--	12-17 07:10:27	12-17 07:10:27
automount	94	94	N	N	0	--	12-17 07:10:27	12-17 07:10:28
knotify	111	111	N	N	0	--	12-17 07:10:28	12-17 07:10:28
comsh	121	121	N	N	0	--	12-17 07:10:30	12-17 07:10:30
knotify	152	152	N	N	0	--	12-17 07:10:31	12-17 07:10:31
autocfgd	155	155	N	N	0	--	12-17 07:10:31	12-17 07:10:31
pkg_update	122	122	N	N	0	--	12-17 07:10:30	12-17 07:10:31

表1-9 display process log 命令显示信息描述表

字段	描述
Process	用户态进程名
JobID	用户态进程任务编号
PID	用户态进程编号

字段	描述
Abort	是否异常退出： <ul style="list-style-type: none"> <li>Y 表示异常退出</li> <li>N 表示正常退出</li> </ul>
Core	是否产生core文件 <ul style="list-style-type: none"> <li>Y 表示产生</li> <li>N 表示未产生</li> </ul>
Exit	进程退出码，取值为： <ul style="list-style-type: none"> <li>数字表示进程退出码</li> <li>--表示无退出码，进程被信号关闭</li> </ul>
Kill	关闭进程的信号，取值为： <ul style="list-style-type: none"> <li>数字表示关闭进程的信号的数值</li> <li>--表示没有关闭信号，进程主动退出，并非被信号关闭</li> </ul>
StartTime	用户态进程创建时间
EndTime	用户态进程结束时间

### 1.1.12 display process memory

**display process memory** 命令用来显示所有用户态进程的代码段、数据段以及堆栈等的内存使用信息。

#### 【命令】

**display process memory** [ **slot** *slot-number* [ **cpu** *cpu-number* ] ]

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

**slot** *slot-number*: 表示设备在 IRF 中的成员编号。不指定该参数时，表示主设备。

**cpu** *cpu-number*: 表示 CPU 的编号。

#### 【使用指导】

用户态进程启动时，会向系统申请 **Text**、**Data**、**Stack** 和 **Dynamic** 类型的内存：

- Text 类型的内存用来存放用户态进程的代码。
- Data 类型的内存用来存放用户态进程的数据。
- Stack 内存指的是栈内存，一般存放临时数据。

- Dynamic 类型的内存指的是堆内存（heap），由系统根据用户态进程运行需要进行动态分配（malloc）和释放（free），可使用 **display process memory heap** 命令显示 Dynamic 类型内存的详细信息。

【举例】

# 显示所有用户态进程的内存使用信息。

```
<Sysname> display process memory

  JID      Text      Data      Stack      Dynamic      Name
  ---
   1       384      1800       16         36        scmd
   2         0         0         0          0    [kthreadd]
   3         0         0         0          0    [ksoftirqd/0]
   4         0         0         0          0    [watchdog/0]
   5         0         0         0          0    [events/0]
   6         0         0         0          0    [khelper]
  29         0         0         0          0    [kblockd/0]
  49         0         0         0          0    [vzmond]
  52         0         0         0          0    [pdflush]

---- More ----
```

表1-10 display process memory 命令显示信息描述表

字段	描述
JID	任务编号。用于唯一标识一个用户态进程，该编号不会随着用户态进程的重启而改变
Text	用户态进程占用的代码段大小，单位为KB（内核线程该项大小为0）
Data	用户态进程占用的数据段大小，单位为KB（内核线程该项大小为0）
Stack	用户态进程占用的堆栈大小，单位为KB（内核线程该项大小为0）
Dynamic	用户态进程动态申请内存大小，单位为KB（内核线程该项大小为0）
Name	用户态进程名称（如果用户态进程名称带有“[]”标记，则表示该进程为内核线程）

【相关命令】

- display process memory heap**
- display process memory heap address**
- display process memory heap size**

1.1.13 display process memory heap

**display process memory heap** 命令用来显示指定用户态进程的堆内存统计信息。

【命令】

```
display process memory heap job job-id [ verbose ] [ slot slot-number [ cpu
cpu-number ] ]
```

【视图】

任意视图

## 【缺省用户角色】

network-admin  
network-operator

## 【参数】

**job job-id**: 任务编号，用于唯一标识一个用户态进程，该编号不会随着用户态进程的重启而改变。取值范围为 1~2147483647。

**verbose**: 显示内存详细统计信息。不指定该参数时，显示内存概要统计信息。

**slot slot-number**: 表示设备在 IRF 中的成员编号。不指定该参数时，表示主设备。

**cpu cpu-number**: 表示 CPU 的编号。

## 【使用指导】

系统的堆内存由固定大小（比如 **size=16** 字节、**size=64** 字节等）的内存块构成，用于存放用户态进程运行过程中需要用到数据或者中间变量。当用户态进程启动时，系统会根据用户态进程运行需要，给用户态进程动态分配堆内存。用户态进程的堆内存信息可使用 **display process memory heap** 命令显示。

每个内存块都有地址，该地址用十六进制数表示，可通过 **display process memory heap size** 命令显示。用户使用内存块的地址可以访问内存块，获取内存块的内容，内存块的内容可通过 **display process memory heap address** 命令显示。

## 【举例】

# 显示 job 1 的堆内存概要统计信息。

```
<Sysname> display process memory heap job 1
Total virtual memory heap space(in bytes) : 2228224
Total physical memory heap space(in bytes) : 262144
Total allocated memory(in bytes)          : 161576
```

# 显示 job 1 的堆内存详细统计信息。

```
<Sysname> display process memory heap job 1 verbose
Heap usage:
Size      Free      Used      Total      Free Ratio
16         8         52         60         13%
64         3        1262       1265        0.2%
128        2         207        209         1%
512        3         55         58         5.1%
4096       3         297        300         1%
8192       1         19         20          5%
81920      0          1          1          0%
Summary:
Total virtual memory heap space (in bytes) : 2293760
Total physical memory heap space (in bytes) : 58368
Total allocated memory (in bytes)          : 42368
```

以上显示信息表明：job 1 分得 size 大小 16 字节的内存块 60 个（已用 52 个，还有 8 个未使用），size 大小为 64 字节的内存块 1265 个（已用 1262 个，还有 3 个未使用），以此类推。

表1-11 display process memory heap 命令显示信息描述表

命令字	功能描述
Total virtual memory heap space(in bytes)	虚拟堆内存总大小，单位为字节
Total physical memory heap space(in bytes)	物理堆内存总大小，单位为字节
Total allocated memory(in bytes)	任务已使用的堆内存大小，单位为字节
Size	内存块大小，单位为字节
Free	空闲的内存块个数
Used	已使用的内存块个数
Total	指定大小内存块总个数，为Free和Used之和
Free Ratio	Free与Total的比率，可以反映这种大小内存块的碎片情况

#### 【相关命令】

- **display process memory**
- **display process memory heap address**
- **display process memory heap size**

#### 1.1.14 display process memory heap address

**display process memory heap address** 命令用来显示从指定地址开始的内存空间的内容。

#### 【命令】

**display process memory heap job** *job-id* **address** *starting-address* **length** *memory-length* [ **slot** *slot-number* [ **cpu** *cpu-number* ] ]

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

**job** *job-id*: 任务编号，用于唯一标识一个用户态进程，该编号不会随着用户态进程的重启而改变，取值范围为 1~2147483647。

**address** *starting-address*: 内存块的起始地址。

**length** *memory-length*: 内存的长度，取值范围为 1~1024，单位为字节。

**slot** *slot-number*: 表示设备在 IRF 中的成员编号。不指定该参数时，表示主设备。

**cpu** *cpu-number*: 表示 CPU 的编号。



## 【使用指导】

当用户态进程运行异常时，使用该命令可以帮助设备维护人员诊断和定位问题。

## 【举例】

# 显示 job 1 从地址 0xb7e30580 开始，长度为 128 字节的内存空间的内容。

```
<Sysname> display process memory heap job 1 address b7e30580 length 128
B7E30580:  14 00 EF FF 00 00 00 00 E4 39 E2 B7 7C 05 E3 B7  .....9..|...
B7E30590:  14 00 EF FF 2F 73 62 69 6E 2F 73 6C 62 67 64 00  ..../sbin/slbgd.
B7E305A0:  14 00 EF FF 00 00 00 00 44 3B E2 B7 8C 05 E3 B7  .....Di.....
B7E305B0:  14 00 EF FF 2F 73 62 69 6E 2F 6F 73 70 66 64 00  ..../sbin/ospfd.
B7E305C0:  14 00 EF FF 00 00 00 00 A4 3C E2 B7 AC 05 E3 B7  .....<.....
B7E305D0:  14 00 EF FF 2F 73 62 69 6E 2F 6D 73 74 70 64 00  ..../sbin/mstpd.
B7E305E0:  14 00 EF FF 00 00 00 00 04 3E E2 B7 CC 05 E3 B7  .....>.....
B7E305F0:  14 00 EF FF 2F 73 62 69 6E 2F 6E 74 70 64 00 00  ..../sbin/ntpd..
```

## 【相关命令】

- **display process memory heap**
- **display process memory heap size**

### 1.1.15 display process memory heap size

**display process memory heap size** 命令用来显示指定大小已使用内存块的地址。

## 【命令】

```
display process memory heap job job-id size memory-size [ offset offset-size ]
[ slot slot-number [ cpu cpu-number ] ]
```

## 【视图】

任意视图

## 【缺省用户角色】

network-admin  
network-operator

## 【参数】

**job** *job-id*: 任务编号，用于唯一标识一个用户态进程，该编号不会随着用户态进程的重启而改变，取值范围为 1~2147483647。

**size** *memory-size*: 内存块大小，取值范围为 1~4294967295。

**offset** *offset-size*: 要查询的内存块的偏移，取值范围为 0~4294967295，缺省值为 128。比如，系统给 job 1 分配了 size 为 16 字节的内存块 100 个，用户态进程当前已用了 66 个，如果执行命令 **display process memory heap job 1 size 16 offset 50**，则会显示该用户态进程第 51 到第 66 个 size 为 16 字节的内存块的地址。

**slot** *slot-number*: 表示设备在 IRF 中的成员编号。不指定该参数时，表示主设备。

**cpu** *cpu-number*: 表示 CPU 的编号。

### 【使用指导】

该命令显示的地址为十六进制格式，使用该地址，通过 **display process memory heap address** 命令可以显示该地址内存的具体内容。

### 【举例】

# 显示 job 1 已使用的 size 大小为 16 字节的内存块的地址。

```
<Sysname> display process memory heap job 1 size 16
0xb7e300c0 0xb7e300d0 0xb7e300e0 0xb7e300f0
0xb7e30100 0xb7e30110 0xb7e30120 0xb7e30130
0xb7e30140 0xb7e30150 0xb7e30160 0xb7e30170
0xb7e30180 0xb7e30190 0xb7e301a0 0xb7e301b0
0xb7e301c0 0xb7e301d0 0xb7e301e0 0xb7e301f0
0xb7e30200 0xb7e30210 0xb7e30220 0xb7e30230
```

# 显示 job 1 已使用的 size 大小为 16 字节的内存块的地址，从第 5 个已使用内存块开始显示。

```
<Sysname> display process memory heap job 1 size 16 offset 4
0xb7e30100 0xb7e30110 0xb7e30120 0xb7e30130
0xb7e30140 0xb7e30150 0xb7e30160 0xb7e30170
0xb7e30180 0xb7e30190 0xb7e301a0 0xb7e301b0
0xb7e301c0 0xb7e301d0 0xb7e301e0 0xb7e301f0
0xb7e30200 0xb7e30210 0xb7e30220 0xb7e30230
```

### 【相关命令】

- **display process memory heap**
- **display process memory heap address**

## 1.1.16 exception filepath

**exception filepath** 命令用来配置 core 文件的保存路径。

**undo exception filepath** 命令用来将 core 文件的保存路径配置为空。

### 【命令】

```
exception filepath directory
undo exception filepath directory
```

### 【缺省情况】

Core 文件的保存路径为 flash:。

### 【视图】

用户视图

### 【缺省用户角色】

network-admin

### 【参数】

*directory*: 表示 core 文件的保存路径，只能为存储介质的根目录。

### 【使用指导】

本命令配置成功后，设备会将生成的 **core** 文件存放到当前主设备上指定存储介质根目录下的 **core** 文件夹下。如果存储介质根目录下没有 **core** 文件夹，则会先创建 **core** 文件夹，再保存 **core** 文件。  
当设备上有不同类型存储介质的时候，可使用该命令修改 **core** 文件的保存路径。  
需要注意的是，当 **core** 文件的保存路径为空或无法正常访问时，系统将无法保存 **core** 文件。

### 【举例】

```
# 配置 core 文件的保存路径。  
<Sysname> exception filepath flash:/
```

### 【相关命令】

- **display exception filepath**
- **process core**

## 1.1.17 monitor kernel deadlock action

**monitor kernel deadlock action** 命令用来配置内核线程死循环后系统执行的操作。  
**undo monitor kernel deadlock action** 命令用来恢复缺省情况。

### 【命令】

```
monitor kernel deadlock action { reboot | record-only } [ slot slot-number  
[ cpu cpu-number ] ]  
undo monitor kernel deadlock action [ slot slot-number [ cpu cpu-number ] ]
```

### 【缺省情况】

系统检测到内核线程死循环后，执行的操作为 **reboot**。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**reboot**: 表示检测到内核线程死循环后，先记录日志再自动重启指定 Slot、CPU。  
**record-only**: 表示检测到内核线程死循环后，只记录日志不自动重启。  
**slot slot-number**: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。  
**cpu cpu-number**: 表示 CPU 的编号。

### 【使用指导】

通常情况下，请不要配置本命令。如果需要配置，请在工程师的指导下进行，以免引起系统异常。

### 【举例】

```
# 配置内核线程被判定为死循环后重启 slot 1。  
<Sysname> system-view  
[Sysname] monitor kernel deadlock action reboot slot 1
```

## 【相关命令】

- `display kernel deadlock configuration`
- `monitor kernel deadlock enable`

### 1.1.18 monitor kernel deadlock enable

`monitor kernel deadlock enable` 命令用来开启内核线程死循环检测功能。

`undo monitor kernel deadlock enable` 命令用来关闭内核线程死循环检测功能。

## 【命令】

`monitor kernel deadlock enable [ slot slot-number [ cpu cpu-number [ core core-number<1-64> ] ] ]`

`undo monitor kernel deadlock enable [ slot slot-number [ cpu cpu-number ] ]`

## 【缺省情况】

内核线程死循环检测功能处于开启状态。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**slot slot-number**: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。

**cpu cpu-number**: 表示 CPU 的编号。

**core core-number<1-64>**: 表示 CPU 核的编号。不指定该参数时，表示当前 CPU 上的所有核。<1-64>表示前面的参数最多可以输入 64 次。

## 【使用指导】

在内核态空间中，所有资源都是共享的，多个内核线程之间通过任务调度协调工作。如果某个内核线程长时间一直占用 CPU，就会导致其它内核线程获取不到运行机会，整个系统挂死，这种现象称为死循环。

开启内核线程死循环检测功能后，如果系统发现某内核线程在指定时间内一直占用 CPU，则判定该内核线程为死循环，并记录一条死循环信息供管理员查询。

通常情况下，请不要配置本命令。如果确实需要配置，请在工程师的指导下进行，以免引起系统异常。

## 【举例】

# 开启内核线程死循环检测功能。

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

```
[Sysname] monitor kernel deadlock enable
```

## 【相关命令】

- `display kernel deadlock`
- `display kernel deadlock configuration`

- `monitor kernel deadlock action`
- `monitor kernel deadlock exclude-thread`
- `monitor kernel deadlock time`

### 1.1.19 monitor kernel deadlock exclude-thread

`monitor kernel deadlock exclude-thread` 命令用来配置不检测指定内核线程是否发生了死循环。

`undo monitor kernel deadlock exclude-thread` 命令用来恢复对指定内核线程是否发生了死循环进行检测。

#### 【命令】

```
monitor kernel deadlock exclude-thread tid [ slot slot-number [ cpu
cpu-number ] ]
undo monitor kernel deadlock exclude-thread [ tid ] [ slot slot-number [ cpu
cpu-number ] ]
```

#### 【缺省情况】

开启内核线程死循环检测功能后，系统会监控所有内核线程是否发生了死循环。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

`tid`: 表示内核线程编号，用于唯一标识一个内核线程，取值范围为 1~2147483647。不指定该参数时，表示恢复到缺省情况。

`slot slot-number`: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。

`cpu cpu-number`: 表示 CPU 的编号。

#### 【使用指导】

多次执行该命令，可以配置对多个内核线程不进行检测，最多可以配置 128 个。

通常情况下，请不要配置本命令。如果需要配置，请在工程师的指导下进行，以免引起系统异常。

#### 【举例】

# 对编号为 15 的内核线程不进行死循环检测。

```
<Sysname> system-view
[Sysname]monitor kernel deadlock exclude-thread 15
```

#### 【相关命令】

- `display kernel deadlock configuration`
- `display kernel deadlock`
- `monitor kernel deadlock enable`

### 1.1.20 monitor kernel deadlock time

**monitor kernel deadlock time** 命令用来配置判定内核线程是否死循环的时长。

**undo monitor kernel deadlock time** 命令用来恢复缺省情况。

#### 【命令】

**monitor kernel deadlock time** *time* [ **slot** *slot-number* [ **cpu** *cpu-number* ] ]

**undo monitor kernel deadlock time** [ **slot** *slot-number* [ **cpu** *cpu-number* ] ]

#### 【缺省情况】

当某内核线程连续运行超过 20 秒钟，则判定为死循环。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**time** *time*: 表示内核线程死循环判定时长，取值范围为 1~65535，单位为秒。

**slot** *slot-number*: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。

**cpu** *cpu-number*: 表示 CPU 的编号。

#### 【使用指导】

开启内核线程检测功能后，如果某内核线程持续运行指定时间，则认为该内核线程已经死循环。

通常情况下，请不要配置本命令。如果需要配置，请在工程师的指导下进行，以免引起系统异常。

#### 【举例】

# 配置当某内核线程连续运行超过 8 秒钟，则判定为死循环。

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

```
[Sysname] monitor kernel deadlock time 8
```

#### 【相关命令】

- **display kernel deadlock configuration**
- **display kernel deadlock**
- **monitor kernel deadlock enable**

### 1.1.21 monitor kernel starvation enable

**monitor kernel starvation enable** 命令用来开启内核线程饿死检测功能。

**undo monitor kernel starvation enable** 命令用来关闭内核线程饿死检测功能。

#### 【命令】

**monitor kernel starvation enable** [ **slot** *slot-number* [ **cpu** *cpu-number* ] ]

**undo monitor kernel starvation enable** [ **slot** *slot-number* [ **cpu** *cpu-number* ] ]

#### 【缺省情况】

内核线程饿死检测功能处于关闭状态。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**slot** *slot-number*: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。

**cpu** *cpu-number*: 表示 CPU 的编号。

## 【使用指导】

如果内核线程本身的触发条件没有达到，会导致该内核线程在一段时间内一直得不到调度，这种现象称为饿死。

开启内核线程饿死检测功能后，当系统检测到某内核线程饿死时，会记录一条饿死信息供管理员查询。

内核线程饿死并不会影响整个系统的运行，当触发条件达到，处于饿死状态的内核线程会自动执行。通常情况下，请不要配置本命令。如果确实需要配置，请在工程师的指导下进行，以免引起系统异常。

## 【举例】

# 开启内核线程饿死检测功能。

```
<Sysname> system-view
[Sysname] monitor kernel starvation enable
```

## 【相关命令】

- **display kernel starvation configuration**
- **display kernel starvation**
- **monitor kernel starvation time**
- **monitor kernel starvation exclude-thread**

### 1.1.22 monitor kernel starvation exclude-thread

**monitor kernel starvation exclude-thread** 命令用来配置不检测指定内核线程是否发生了饿死。

**undo monitor kernel starvation exclude-thread** 命令用来恢复对指定内核线程是否发生了饿死进行检测。

## 【命令】

```
monitor kernel starvation exclude-thread tid [ slot slot-number [ cpu cpu-number ] ]
undo monitor kernel starvation exclude-thread [ tid ] [ slot slot-number [ cpu cpu-number ] ]
```

## 【缺省情况】

开启内核线程饿死检测功能后，会监控所有内核线程是否发生了饿死。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**tid**: 表示内核线程编号，用于唯一标识一个内核线程，取值范围为 1~2147483647。不指定该参数时，表示恢复到缺省情况。

**slot slot-number**: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。

**cpu cpu-number**: 表示 CPU 的编号。

### 【使用指导】

多次执行该命令，可以配置对多个内核线程不进行检测，最多可以配置 128 个。

通常情况下，请不要配置本命令。如果确实需要配置，请在工程师的指导下进行，以免引起系统异常。

### 【举例】

# 对编号为 15 的内核线程不进行饿死检测。

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

```
[Sysname] monitor kernel starvation exclude-thread 15
```

### 【相关命令】

- **display kernel starvation**
- **display kernel starvation configuration**
- **monitor kernel starvation enable**

## 1.1.23 monitor kernel starvation time

**monitor kernel starvation time** 命令用来配置判定内核线程是否饿死的时长。

**undo monitor kernel starvation time** 命令用来恢复缺省情况。

### 【命令】

```
monitor kernel starvation time time [ slot slot-number [ cpu cpu-number ] ]
```

```
undo monitor kernel starvation time [ slot slot-number [ cpu cpu-number ] ]
```

### 【缺省情况】

当某内核线程在 120 秒内一直没有运行，则认为该内核线程被饿死。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**time time**: 表示内核线程饿死判定时长，取值范围为 1~65535，单位为秒。



**slot slot-number**: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。

**cpu cpu-number**: 表示 CPU 的编号。

#### 【使用指导】

通常情况下，请不要配置本命令。如果需要配置，请在工程师的指导下进行，以免引起系统异常。

#### 【举例】

# 配置当内核线程在 120 秒内一直没有运行，则认为该内核线程被饿死。

```
<Sysname> system-view
[Sysname] monitor kernel starvation time 120
```

#### 【相关命令】

- **display kernel starvation**
- **display kernel starvation configuration**
- **monitor kernel starvation enable**

### 1.1.24 monitor process

**monitor process** 命令用来显示进程的统计信息。

#### 【命令】

```
monitor process [ dumbtty ] [ iteration number ] [ slot slot-number [ cpu
cpu-number ] ]
```

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**dumbtty**: 以哑终端方式显示进程统计信息（即屏幕不支持定时刷新统计信息）。指定该参数时，全部进程的统计信息以 CPU 使用率降序排列输出到屏幕上；不指定该参数时，统计信息以交互模式显示，缺省情况下按 CPU 占用率降序显示前 10 个进程的统计信息，且每隔 5 秒刷新一次。

**iteration number**: 表示进程统计信息的显示次数，取值范围为 1~4294967295。指定 **dumbtty** 参数时，**number** 的缺省值为 1；不指定 **dumbtty** 且不配置 **number** 参数时，表示显示次数没有限制，统计信息会每隔 5 秒刷新一次，一直显示。

**slot slot-number**: 表示设备在 IRF 中的成员编号。不指定该参数时，表示主设备。

**cpu cpu-number**: 表示 CPU 的编号。

#### 【使用指导】

不指定 **dumbtty** 参数的情况下，统计信息将以交互模式显示。在交互模式下，系统可以根据用户输入的交互命令字来调整显示方式或直接终止进程。在用户输入交互命令字之前，系统会自动计算可显示的进程个数，超出屏幕范围的进程不会显示。

在交互模式下可以使用的交互命令字及对应的功能请参见 [表 1-12](#)。

表1-12 monitor process 命令支持的交互命令字描述表

命令字	功能描述
?或h	帮助信息，显示可用的交互式命令字
1	各物理CPU状态的显示开关。比如： 1. 输入 1，分别显示各物理 CPU 的参数值 2. 再次输入 1，显示所有 CPU 的参数的平均值 3. 第三次输入 1，又分别显示各物理 CPU 的参数值 4. 如此循环 缺省情况下，显示所有CPU的参数的平均值
c	按CPU占用率降序排列，缺省情况下采用降序排列
d	配置统计信息的更新时间间隔，取值范围为1~2147483647秒，缺省值为5
f	按进程打开的文件句柄数降序排列
k	终止一个任务，此命令会影响系统运行，请谨慎使用
l	刷新屏幕
m	按进程使用内存大小降序排列
n	改变显示的进程个数，取值范围为0~2147483647（缺省值为10，0表示不作限制）；超过屏幕范围时，仍只显示一屏内可容纳的进程个数
q	退出交互模式
t	按进程最近一次启动后的运行时间降序排列
<	排序项向左移动一列
>	排序项向右移动一列

【举例】

# 以哑终端方式显示进程统计信息。（使用该方式显示时，系统会一次显示所有进程的统计信息，并且不支持定时刷新，显示完毕后，会退回到命令视图）

```
<Sysname> monitor process dumbtty
76 processes; 103 threads; 687 fds
Thread states: 1 running, 102 sleeping, 0 stopped, 0 zombie
CPU states: 77.16% idle, 0.00% user, 14.96% kernel, 7.87% interrupt
Memory: 496M total, 341M available, page size 4K
  JID   PID   PRI  State  FDs    MEM  HH:MM:SS   CPU   Name
  1047  1047  120   R      9    1420K  00:02:23  13.53%  diagd
    1     1   120   S     17    1092K  00:00:20   7.61%  scmd
  1000  1000  115   S      0      0K    00:00:09   0.84%  [sock/1]
  1026  1026  120   S     20   26044K  00:00:05   0.84%  syslogd
    2     2   115   S      0      0K    00:00:00   0.00%  [kthreadd]
    3     3    99   S      0      0K    00:00:00   0.00%  [migration/0]
    4     4   115   S      0      0K    00:00:06   0.00%  [ksoftirqd/0]
    5     5    99   S      0      0K    00:00:00   0.00%  [watchdog/0]
    6     6   115   S      0      0K    00:00:01   0.00%  [events/0]
```

```

      7      7 115    S    0      0K 00:00:00  0.00% [khelper]
4797  4797 120    S    8 28832K 00:00:02  0.00% comsh
5117  5117 120    S    8  1496K 00:00:00  0.00% top

```

<Sysname>

# 以哑终端方式显示进程统计信息，并且执行一次命令显示两次统计结果。

<Sysname> monitor process dumbtty iteration 2

76 processes; 103 threads; 687 fds

Thread states: 1 running, 102 sleeping, 0 stopped, 0 zombie

CPU states: 44.84% idle, 0.51% user, 39.17% kernel, 15.46% interrupt

Memory: 496M total, 341M available, page size 4K

JID	PID	PRI	State	FDs	MEM	HH:MM:SS	CPU	Name
1047	1047	120	R	9	1420K	00:02:30	37.11%	diagd
1	1	120	S	17	1092K	00:00:21	11.34%	scmd
1000	1000	115	S	0	0K	00:00:09	2.06%	[sock/1]
1026	1026	120	S	20	26044K	00:00:05	1.54%	syslogd
1027	1027	120	S	12	9280K	00:01:12	1.03%	devd
4	4	115	S	0	0K	00:00:06	0.51%	[ksoftirqd/0]
1009	1009	115	S	0	0K	00:00:08	0.51%	[karp/1]
1010	1010	115	S	0	0K	00:00:13	0.51%	[kND/1]
5373	5373	120	S	8	1496K	00:00:00	0.51%	top
2	2	115	S	0	0K	00:00:00	0.00%	[kthreadd]
3	3	99	S	0	0K	00:00:00	0.00%	[migration/0]
5	5	99	S	0	0K	00:00:00	0.00%	[watchdog/0]
6	6	115	S	0	0K	00:00:01	0.00%	[events/0]
7	7	115	S	0	0K	00:00:00	0.00%	[khelper]
4796	4796	120	S	11	2744K	00:00:00	0.00%	login
4797	4797	120	S	8	28832K	00:00:03	0.00%	comsh

// 5 秒钟后，系统会自动统计一次，并显示统计信息如下。（相当于执行了两次 **monitor process dumbtty**，两次执行的时间间隔为 5 秒）

76 processes; 103 threads; 687 fds

Thread states: 1 running, 102 sleeping, 0 stopped, 0 zombie

CPU states: 78.71% idle, 0.16% user, 14.86% kernel, 6.25% interrupt

Memory: 496M total, 341M available, page size 4K

JID	PID	PRI	State	FDs	MEM	HH:MM:SS	CPU	Name
1047	1047	120	R	9	1420K	00:02:31	14.25%	diagd
1	1	120	S	17	1092K	00:00:21	4.25%	scmd
1027	1027	120	S	12	9280K	00:01:12	1.29%	devd
1000	1000	115	S	0	0K	00:00:09	0.37%	[sock/1]
5373	5373	120	S	8	1500K	00:00:00	0.37%	top
6	6	115	S	0	0K	00:00:01	0.18%	[events/0]
1009	1009	115	S	0	0K	00:00:08	0.18%	[karp/1]
1010	1010	115	S	0	0K	00:00:13	0.18%	[kND/1]
4795	4795	120	S	11	2372K	00:00:01	0.18%	telnetd
2	2	115	S	0	0K	00:00:00	0.00%	[kthreadd]
3	3	99	S	0	0K	00:00:00	0.00%	[migration/0]
4	4	115	S	0	0K	00:00:06	0.00%	[ksoftirqd/0]
5	5	99	S	0	0K	00:00:00	0.00%	[watchdog/0]

```

      7      7 115    S      0      0K 00:00:00  0.00% [khelper]
4796  4796 120    S     11  2744K 00:00:00  0.00% login
4797  4797 120    S      8 28832K 00:00:03  0.00% comsh

```

<Sysname>

# 以交互方式显示进程统计信息。

<Sysname> monitor process

76 processes; 103 threads; 687 fds

Thread states: 1 running, 102 sleeping, 0 stopped, 0 zombie

CPU states: 78.98% idle, 0.16% user, 14.57% kernel, 6.27% interrupt

Memory: 496M total, 341M available, page size 4K

JID	PID	PRI	State	FDs	MEM	HH:MM:SS	CPU	Name
1047	1047	120	R	9	1420K	00:02:39	14.13%	diagd
1	1	120	S	17	1092K	00:00:23	3.98%	scmd
1027	1027	120	S	12	9280K	00:01:13	1.44%	devd
1000	1000	115	S	0	0K	00:00:09	0.36%	[sock/1]
1009	1009	115	S	0	0K	00:00:09	0.36%	[karp/1]
4	4	115	S	0	0K	00:00:06	0.18%	[ksoftirqd/0]
1010	1010	115	S	0	0K	00:00:13	0.18%	[kND/1]
4795	4795	120	S	11	2372K	00:00:01	0.18%	telnetd
5491	5491	120	S	8	1500K	00:00:00	0.18%	top
2	2	115	S	0	0K	00:00:00	0.00%	[kthreadd]

以上信息会每隔 5 秒刷新一次。

- 输入 “h” 或 “?”，将显示如下帮助信息。

Help for interactive commands:

```

?,h    Show the available interactive commands
l      Toggle SMP view: 'l' single/separate states
c      Sort by the CPU field(default)
d      Set the delay interval between screen updates
f      Sort by number of open files
k      Kill a job
l      Refresh the screen
m      Sort by memory used
n      Set the maximum number of processes to display
q      Quit the interactive display
t      Sort by run time of processes since last restart
<      Move sort field to the next left column
>      Move sort field to the next right column

```

Press any key to continue

- 输入 “d” 后，根据出现的提示如果输入 “3”，则统计信息将会每隔 3 秒更新一次。

Enter the delay interval between updates(1~2147483647): 3

- 输入 “n” 后，根据出现的提示如果输入 “5”，则显示的进程数目将会变为 5 个。

Enter the max number of processes to display(0 means unlimited): 5

87 processes; 113 threads; 735 fds

Thread states: 2 running, 111 sleeping, 0 stopped, 0 zombie

CPU states: 86.57% idle, 0.83% user, 11.74% kernel, 0.83% interrupt

Memory: 755M total, 414M available, page size 4K

JID	PID	PRI	State	FDs	MEM	HH:MM:SS	CPU	Name
864	864	120	S	24	27020K	00:00:43	8.95%	syslogd
1173	1173	120	R	24	2664K	00:00:01	2.37%	top
866	866	120	S	18	10276K	00:00:09	0.69%	devd
1	1	120	S	16	1968K	00:00:04	0.41%	scmd
881	881	120	S	8	2420K	00:00:07	0.41%	diagd

- 输入“f”，统计信息将以打开的文件句柄数降序输出（c、m、t命令字类似）。

```
87 processes; 113 threads; 735 fds
Thread states: 1 running, 112 sleeping, 0 stopped, 0 zombie
CPU states: 90.66% idle, 0.88% user, 5.77% kernel, 2.66% interrupt
Memory: 755M total, 414M available, page size 4K
```

JID	PID	PRI	State	FDs	MEM	HH:MM:SS	CPU	Name
862	862	120	S	61	5384K	00:00:01	0.00%	dbmd
905	905	120	S	35	2464K	00:00:02	0.00%	ipbased
863	863	120	S	31	1956K	00:00:00	0.00%	had
884	884	120	S	31	30600K	00:00:00	0.00%	lsmd
889	889	120	S	29	61592K	00:00:00	0.00%	routed

- 输入“k”后，根据出现的提示如果输入884，将会终止此JID对应的任务“lsmd”。

```
Enter the JID to kill: 884
84 processes; 107 threads; 683 fds
Thread states: 1 running, 106 sleeping, 0 stopped, 0 zombie
CPU states: 59.03% idle, 1.92% user, 37.88% kernel, 1.15% interrupt
Memory: 755M total, 419M available, page size 4K
```

JID	PID	PRI	State	FDs	MEM	HH:MM:SS	CPU	Name
862	862	120	S	56	5384K	00:00:01	0.00%	dbmd
905	905	120	S	35	2464K	00:00:02	0.00%	ipbased
863	863	120	S	30	1956K	00:00:00	0.00%	had
889	889	120	S	29	61592K	00:00:00	0.00%	routed
1160	1160	120	S	28	23096K	00:00:01	0.19%	sshd

- 输入“q”，将退出交互模式。

表1-13 monitor process 命令输出信息描述表

字段	描述
84 processes; 107 threads; 683 fds	系统的进程总数，线程总数，文件句柄总数
Thread states: 1 running, 102 sleeping, 0 stopped, 0 zombie	线程状态：处于running状态的线程数，处于sleeping（包括interruptible sleep和uninterruptible sleep）状态的线程数，处于stopped状态的线程数，处于zombie状态的线程数
CPU states	CPU状态：空闲率，用户态占用率，内核态占用率，中断占用率
Memory	内存状态：总量，可用内存数，page大小，单位为KB
JID	任务编号（用于唯一标识一个进程，该编号不会随着进程的重启而改变）
PID	进程编号
PRI	进程优先级

字段	描述
State	进程状态，可能的取值为： <ul style="list-style-type: none"> <li>• R: running, 运行状态或处于运行队列</li> <li>• S: sleeping, 可中断睡眠状态</li> <li>• T: traced or stopped, 暂停状态</li> <li>• D: uninterruptible sleep, 不可中断睡眠状态</li> <li>• Z: zombie, 僵死状态</li> </ul>
FDs	file descriptions, 进程打开的文件句柄数
MEM	进程所使用的内存大小（内核线程该项显示为0）
HH:MM:SS	进程自最近一次启动以来的运行时间
CPU	进程CPU使用率
Name	进程名称（如果进程名称带有“[]”标记，则表示该进程为内核线程）

### 1.1.25 monitor thread

**monitor thread** 命令用来显示线程的统计信息。

#### 【命令】

```
monitor thread [ dumbtty ] [ iteration number ] [ slot slot-number [ cpu
cpu-number ] ]
```

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**dumbtty**: 以哑终端方式显示线程统计信息（即屏幕不支持定时刷新统计信息）。指定该参数时，全部线程的统计信息以 CPU 使用率降序排列输出到屏幕上。不指定该参数时，统计信息以交互模式显示，缺省情况下按 CPU 占用率降序显示前 10 个线程的统计信息，且每隔 5 秒更新一次。

**iteration number**: 进程统计信息的显示次数，取值范围为 1~4294967295。指定 **dumbtty** 参数时 **number** 的缺省值为 1；不指定 **dumbtty** 且不配置 **number** 参数时，表示显示次数没有限制，统计信息会一直显示。

**slot slot-number**: 表示设备在 IRF 中的成员编号。不指定该参数时，表示主设备。

**cpu cpu-number**: 表示 CPU 的编号。

#### 【使用指导】

不指定 **dumbtty** 参数的情况下，统计信息将以交互模式显示。在交互模式下，系统可以根据用户输入的交互命令字来调整显示方式或直接终止进程。在用户输入交互命令字之前，系统会自动计算可显示的线程个数，超出屏幕范围的线程不会显示。

在交互模式下可以使用的交互命令字及对应的功能请参见 [表 1-14](#)。

表1-14 monitor thread 命令支持的交互命令字描述表

命令字	功能描述
?或h	帮助信息，显示可用的交互式命令字
d	配置统计信息的更新时间间隔，单位为秒，缺省值为5
k	终止一个任务（进程），此命令会影响系统运行，请谨慎使用
l	刷新屏幕
n	改变显示的线程个数，取值为0~2147483647（缺省值为10，0表示不作限制）；超过屏幕范围时，仍只显示一屏内可容纳的线程个数
q	退出交互模式
<	排序项向左移动一列
>	排序项向右移动一列

【举例】

# 以哑终端方式显示线程统计信息。

```
<Sysname> monitor thread dumbtty
84 processes; 107 threads
Thread states: 1 running, 106 sleeping, 0 stopped, 0 zombie
CPU states: 83.19% idle, 1.68% user, 10.08% kernel, 5.04% interrupt
Memory: 755M total, 417M available, page size 4K
  JID   TID   LAST_CPU  PRI   State  HH:MM:SS  MAX   CPU   Name
  1175  1175    0      120    R    00:00:00    1  10.75%  top
    1     1     0      120    S    00:00:06    1   2.68%  scmd
  881   881    0      120    S    00:00:09    1   2.01%  diagd
  776   776    0      120    S    00:00:01    0   0.67%  [DEVd]
  866   866    0      120    S    00:00:11    1   0.67%  devd
    2     2     0      115    S    00:00:00    0   0.00%  [kthreadd]
    3     3     0      115    S    00:00:01    0   0.00%  [ksoftirqd/0]
    4     4     0       99    S    00:00:00    1   0.00%  [watchdog/0]
    5     5     0      115    S    00:00:00    0   0.00%  [events/0]
    6     6     0      115    S    00:00:00    0   0.00%  [khelper]
  796   796    0      115    S    00:00:00    0   0.00%  [kip6fs/1]
```

<Sysname>

# 以交互模式显示线程统计信息。

```
<Sysname> monitor thread
84 processes; 107 threads
Thread states: 1 running, 106 sleeping, 0 stopped, 0 zombie
CPU states: 94.43% idle, 0.76% user, 3.64% kernel, 1.15% interrupt
Memory: 755M total, 417M available, page size 4K
  JID   TID   LAST_CPU  PRI   State  HH:MM:SS  MAX   CPU   Name
  1176  1176    0      120    R    00:00:01    1   3.42%  top
  866   866    0      120    S    00:00:12    1   0.85%  devd
  881   881    0      120    S    00:00:09    1   0.64%  diagd
```

1	1	0	120	S	00:00:06	1	0.42%	scmd
1160	1160	0	120	S	00:00:01	1	0.21%	sshd
2	2	0	115	S	00:00:00	0	0.00%	[kthreadd]
3	3	0	115	S	00:00:01	0	0.00%	[ksoftirqd/0]
4	4	0	99	S	00:00:00	1	0.00%	[watchdog/0]
5	5	0	115	S	00:00:00	0	0.00%	[events/0]
6	6	0	115	S	00:00:00	0	0.00%	[khelper]

- 输入“h”或“?”，帮助信息显示如下：

Help for interactive commands:

```
? ,h      Show the available interactive commands
c         Sort by the CPU field(default)
d         Set the delay interval between screen updates
k         Kill a job
l         Refresh the screen
n         Set the maximum number of threads to display
q         Quit the interactive display
t         Sort by run time of threads since last restart
<         Move sort field to the next left column
>         Move sort field to the next right column
```

Press any key to continue

- 输入“d”后，根据出现的提示如果输入“3”，统计信息将会每隔3秒更新一次。

Enter the delay interval between screen updates (1~2147483647): 3

- 输入“n”后，根据出现的提示如果输入“5”，显示的线程数目将会变为5个。

Enter the max number of threads to display(0 means unlimited): 5

84 processes; 107 threads

Thread states: 1 running, 106 sleeping, 0 stopped, 0 zombie

CPU states: 93.26% idle, 0.99% user, 4.23% kernel, 1.49% interrupt

Memory: 755M total, 417M available, page size 4K

JID	TID	LAST_CPU	PRI	State	HH:MM:SS	MAX	CPU	Name
1176	1176	0	120	R	00:00:02	1	3.71%	top
1	1	0	120	S	00:00:06	1	0.92%	scmd
866	866	0	120	S	00:00:13	1	0.69%	devd
881	881	0	120	S	00:00:10	1	0.69%	diagd
720	720	0	115	D	00:00:01	0	0.23%	[TMTH]

- 输入“k”后，根据出现的提示输入881，将会终止此JID对应的任务diagd。

Enter the JID to kill: 881

83 processes; 106 threads

Thread states: 1 running, 105 sleeping, 0 stopped, 0 zombie

CPU states: 96.26% idle, 0.54% user, 2.63% kernel, 0.54% interrupt

Memory: 755M total, 418M available, page size 4K

JID	TID	LAST_CPU	PRI	State	HH:MM:SS	MAX	CPU	Name
1176	1176	0	120	R	00:00:04	1	1.86%	top
866	866	0	120	S	00:00:14	1	0.87%	devd
1	1	0	120	S	00:00:07	1	0.49%	scmd
730	730	0	0	S	00:00:04	1	0.12%	[DIBC]
762	762	0	120	S	00:00:22	1	0.12%	[MNET]

- 输入“q”，将退出交互模式。



表1-15 monitor thread 命令显示信息描述表

显示项	内容描述
84 processes; 107 threads	系统的进程总数，线程总数
Thread states	线程状态：处于running状态的线程数，处于sleeping（包括interruptible sleep和uninterruptible sleep）状态的线程数，处于stopped状态的线程数，处于zombie状态的线程数
CPU states	CPU状态：空闲率，用户态占用率，内核态占用率，中断占用率
Memory	内存状态：总量，可用内存数，page大小
JID	任务编号，用于唯一标识一个进程，该编号不会随着进程的重启而改变
TID	线程编号
LAST_CPU	线程最近一次被调度所在的CPU的编号
PRI	线程优先级
State	进程状态，可能的取值为： <ul style="list-style-type: none"><li>• R: running，运行状态或处于运行队列</li><li>• S: sleeping，可中断睡眠状态</li><li>• T: traced or stopped，暂停状态</li><li>• D: uninterruptible sleep，不可中断睡眠状态</li><li>• Z: zombie，僵死状态</li></ul>
HH:MM:SS	线程自最近一次启动以来的运行时间
MAX	线程单次调度占用CPU的最长时间，以毫秒为单位
CPU	线程CPU使用率
Name	线程名称（如果线程名称带有“[]”标记，则表示该线程为内核线程）

1.1.26 process core

**process core** 命令用来开启/关闭用户态进程异常时生成 **core** 文件的功能，并配置可生成 **core** 文件的最大个数。

【命令】

```
process core { maxcore value | off } { job job-id | name process-name } [ slot slot-number [ cpu cpu-number ] ]
```

【视图】

用户视图

【缺省情况】

同一用户态进程在首次异常时会生成 **core** 文件，后续异常不再生成 **core** 文件。即 **maxcore** 的最大数值为 1。

### 【缺省用户角色】

network-admin

### 【参数】

**off**: 表示关闭用户态进程异常时生成 **core** 文件的功能。

**maxcore value**: 表示开启用户态进程的 **core** 文件生成功能，并配置能生成的 **core** 文件的最大个数。**value** 表示用户态进程能生成的 **core** 文件的最大个数，取值范围为 1~10，缺省值为 1。

**name process-name**: 用户态进程的名称，为 1~15 个字符的字符串，不区分大小写。**process core** 命令的配置对用户态进程下的所有实例有效。

**job job-id**: 任务 ID，用于唯一标识一个进程，该 ID 不会随着进程的重启而改变，取值范围为 1~2147483647。

**slot slot-number**: 表示设备在 IRF 中的成员编号。不指定该参数时，表示主设备。

**cpu cpu-number**: 表示 CPU 的编号。

### 【使用指导】

开启用户态进程的 **core** 文件生成功能，并配置能生成的 **core** 文件的最大个数后，用户态进程异常重启一次，就会产生一个 **core** 文件并记录用户态进程的异常信息。如果生成的 **core** 文件的数目达到最大值，则不再生成新的 **core** 文件。软件开发和维护人员能够根据 **core** 文件的内容来定位异常的原因和异常的位置。

因为生成的 **core** 文件会占用系统存储资源，如果用户对某些用户态进程的异常退出不关心，可以关闭这些用户态进程的 **core** 文件记录功能。

### 【举例】

# 关闭用户态进程 **routed** 的 **core** 文件生成功能。

```
<Sysname> process core off name routed
```

# 开启用户态进程 **routed** 的 **core** 文件生成功能，并且最多可生成 5 个 **core** 文件。

```
<Sysname> process core maxcore 5 name routed
```

### 【相关命令】

- **display exception context**
- **exception filepath**

## 1.1.27 reset exception context

**reset exception context** 命令用来清除用户态进程异常时记录的上下文信息。

### 【命令】

```
reset exception context [ slot slot-number [ cpu cpu-number ] ]
```

### 【视图】

用户视图

### 【缺省用户角色】

network-admin

### 【参数】

**slot slot-number:** 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。  
**cpu cpu-number:** 表示 CPU 的编号。

### 【举例】

# 清除用户态进程异常记录。  
<Sysname> reset exception context

### 【相关命令】

- **display exception context**

## 1.1.28 reset kernel deadlock

**reset kernel deadlock** 命令用来清除内核线程死循环信息。

### 【命令】

**reset kernel deadlock [ slot slot-number [ cpu cpu-number ] ]**

### 【视图】

用户视图

### 【缺省用户角色】

network-admin

### 【参数】

**slot slot-number:** 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。  
**cpu cpu-number:** 表示 CPU 的编号。

### 【举例】

# 清除内核线程死循环信息。  
<Sysname> reset kernel deadlock

### 【相关命令】

- **display kernel deadlock**

## 1.1.29 reset kernel exception

**reset kernel exception** 命令用来清除内核线程的异常信息。

### 【命令】

**reset kernel exception [ slot slot-number [ cpu cpu-number ] ]**

### 【视图】

用户视图

### 【缺省用户角色】

network-admin

### 【参数】

**slot slot-number:** 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。

**cpu** *cpu-number*: 表示 CPU 的编号。

【举例】

# 清除内核线程的异常信息。

```
<Sysname> reset kernel exception
```

【相关命令】

- **display kernel exception**

### 1.1.30 reset kernel reboot

**reset kernel reboot** 命令用来清除内核线程重启信息。

【命令】

```
reset kernel reboot [ slot slot-number [ cpu cpu-number ] ]
```

【视图】

用户视图

【缺省用户角色】

network-admin

【参数】

**slot** *slot-number*: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。

**cpu** *cpu-number*: 表示 CPU 的编号。

【举例】

# 清除内核线程重启信息。

```
<Sysname> reset kernel reboot
```

【相关命令】

- **display kernel reboot**

### 1.1.31 reset kernel starvation

**reset kernel starvation** 命令用来清除内核线程饿死信息。

【命令】

```
reset kernel starvation [ slot slot-number [ cpu cpu-number ] ]
```

【视图】

用户视图

【缺省用户角色】

network-admin

【参数】

**slot** *slot-number*: 表示 IRF 中设备的成员编号。不指定该参数时，表示主设备。

**cpu** *cpu-number*: 表示 CPU 的编号。

### 【举例】

# 清除内核线程饿死信息。

```
<Sysname> reset kernel starvation
```

### 【相关命令】

- `display kernel starvation`

# 目 录

1 端口镜像.....	1-1
1.1 端口镜像配置命令.....	1-1
1.1.1 display mirroring-group .....	1-1
1.1.2 mirroring-group .....	1-2
1.1.3 mirroring-group mirroring-port (interface view).....	1-3
1.1.4 mirroring-group mirroring-port (system view).....	1-4
1.1.5 mirroring-group monitor-egress.....	1-5
1.1.6 mirroring-group monitor-port (interface view) .....	1-6
1.1.7 mirroring-group monitor-port (system view) .....	1-7
1.1.8 mirroring-group reflector-port.....	1-8
1.1.9 mirroring-group remote-probe vlan .....	1-9
2 流镜像.....	2-1
2.1 流镜像配置命令.....	2-1
2.1.1 mirror-to cpu.....	2-1
2.1.2 mirror-to interface .....	2-1

# 1 端口镜像

## 1.1 端口镜像配置命令

### 1.1.1 display mirroring-group

**display mirroring-group** 命令用来显示镜像组的信息。

【命令】

```
display mirroring-group { group-id | all | local | remote-destination |
remote-source }
```

【视图】

任意视图

【缺省用户角色】

network-admin  
network-operator

【参数】

**group-id**: 显示镜像组的信息，取值范围为 1~4。  
**all**: 显示所有镜像组的信息。  
**local**: 显示本地镜像组的信息。  
**remote-destination**: 显示远程目的镜像组的信息。  
**remote-source**: 显示远程源镜像组的信息。

【使用指导】

显示信息的显示顺序按照镜像组的编号顺序排列，显示内容包括镜像组的类型、状态和构成等信息。

【举例】

```
# 显示所有镜像组的信息。
<Sysname> display mirroring-group all
Mirroring group 1:
  Type: Local
  Status: Active
  Mirroring port:
    GigabitEthernet1/0/1  Inbound
  Monitor port: GigabitEthernet1/0/2
```

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

字段	描述
Mirroring group	镜像组的编号

字段	描述
Type	镜像组的类型： <ul style="list-style-type: none"> <li>• Local: 本地镜像组</li> <li>• Remote source: 远程源镜像组</li> <li>• Remote destination: 远程目的镜像组</li> </ul>
Status	镜像组的状态： <ul style="list-style-type: none"> <li>• Active: 表示镜像组已经生效</li> <li>• Incomplete: 表示镜像组没有配完，暂不生效</li> </ul>
Mirroring port	镜像源端口
Monitor port	镜像目的端口
Reflector port	镜像组反射端口
Remote probe VLAN	远程镜像VLAN

### 1.1.2 mirroring-group

**mirroring-group** 命令用来创建镜像组。

**undo mirroring-group** 命令用来删除镜像组。

#### 【命令】

```
mirroring-group group-id { local | remote-destination | remote-source }
undo mirroring-group { group-id | all | local | remote-destination |
remote-source }
```

#### 【缺省情况】

未创建镜像组。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**group-id**: 表示镜像组的编号，取值范围为 1~4。

**local**: 表示本地镜像组。

**remote-destination**: 表示远程目的镜像组。

**remote-source**: 表示远程源镜像组。

**all**: 表示所有镜像组。

#### 【举例】

# 创建本地镜像组 1。

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



```
[Sysname] mirroring-group 1 local
```

### 1.1.3 mirroring-group mirroring-port (interface view)

**mirroring-group mirroring-port** 命令用来配置端口为镜像组的源端口。

**undo mirroring-group mirroring-port** 命令用来恢复缺省情况。

#### 【命令】

```
mirroring-group group-id mirroring-port { both | inbound | outbound }  
undo mirroring-group group-id mirroring-port
```

#### 【缺省情况】

未配置端口为镜像组的源端口。

#### 【视图】

接口视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**group-id**: 表示镜像组的编号，该镜像组必须存在，取值范围为 1~4。

**both**: 表示对端口收发的报文都进行镜像。

**inbound**: 表示仅对端口收到的报文进行镜像。

**outbound**: 表示仅对端口发出的报文进行镜像。

#### 【使用指导】

只能为本地镜像组或远程源镜像组配置源端口，不能为远程目的镜像组配置源端口。

设备不支持将二层聚合接口配置为镜像源端口。

请不要将源端口加入到远程镜像 VLAN 中，否则会影响镜像功能的正常使用。

一个源端口无论作为单向源端口还是双向源端口，都只能加入一个镜像组。

设备仅支持配置一个对源端口的出方向、或双向进行镜像的镜像组。

源端口不能再被用作本镜像组或其他镜像组的反射端口、出端口或目的端口。

#### 【举例】

# 创建本地镜像组 1，配置其源端口为 GigabitEthernet1/0/1，并对该端口收发的报文都进行镜像。

```
<Sysname> system-view  
[Sysname] mirroring-group 1 local  
[Sysname] interface gigabitethernet 1/0/1  
[Sysname-GigabitEthernet1/0/1] mirroring-group 1 mirroring-port both
```

# 创建远程源镜像组 2，配置其源端口为 GigabitEthernet1/0/2，并对该端口收发的报文都进行镜像。

```
<Sysname> system-view  
[Sysname] mirroring-group 2 remote-source  
[Sysname] interface gigabitethernet 1/0/2  
[Sysname-GigabitEthernet1/0/2] mirroring-group 2 mirroring-port both
```

## 【相关命令】

- **mirroring-group**

### 1.1.4 mirroring-group mirroring-port (system view)

**mirroring-group mirroring-port** 命令用来为镜像组配置源端口。

**undo mirroring-group mirroring-port** 命令用来取消镜像组的源端口配置。

## 【命令】

```
mirroring-group group-id mirroring-port interface-list { both | inbound |  
outbound }  
undo mirroring-group group-id mirroring-port interface-list
```

## 【缺省情况】

未为镜像组配置源端口。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**group-id**: 表示镜像组的编号，该镜像组必须存在，取值范围为 1~4。

**interface-list**: 源端口列表，表示一个或多个源端口。表示方式为 **interface-list = { interface-type interface-number [ to interface-type interface-number ] }&<1-8>**。其中，**interface-type interface-number** 为端口类型和端口编号。**&<1-8>**表示前面的参数最多可以输入 8 次。当使用 **to** 参数配置端口范围时，起始端口和终止端口必须是相同 slot 上相同类型的端口，且终止端口的端口编号必须大于等于起始端口的端口编号。

**both**: 表示对端口收发的报文都进行镜像。

**inbound**: 表示仅对端口收到的报文进行镜像。

**outbound**: 表示仅对端口发出的报文进行镜像。

## 【使用指导】

只能为本地镜像组或远程源镜像组配置源端口，不能为远程目的镜像组配置源端口。

设备不支持将二层聚合接口配置为镜像源端口。

请不要将源端口加入到远程镜像 VLAN 中，否则会影响镜像功能的正常使用。

一个源端口无论作为单向源端口还是双向源端口，都只能加入一个镜像组。

设备仅支持配置一个对源端口的出方向、或双向进行镜像的镜像组。

源端口不能再被用作本镜像组或其他镜像组的反射端口、出端口或目的端口。

## 【举例】

# 创建本地镜像组 1，配置其源端口为 GigabitEthernet1/0/1，并对该端口收发的报文都进行镜像。

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

```
[Sysname] mirroring-group 1 local
[Sysname] mirroring-group 1 mirroring-port gigabitethernet 1/0/1 both
# 创建远程源镜像组 2,配置其源端口为 GigabitEthernet1/0/2,并对该端口收发的报文都进行镜像。
<Sysname> system-view
[Sysname] mirroring-group 2 remote-source
[Sysname] mirroring-group 2 mirroring-port gigabitethernet 1/0/2 both
```

## 【相关命令】

- **mirroring-group**

### 1.1.5 mirroring-group monitor-egress

**mirroring-group monitor-egress** 命令用来为远程源镜像组配置出端口。

**undo mirroring-group monitor-egress** 命令用来恢复缺省情况。

## 【命令】

在系统视图下：

```
mirroring-group group-id monitor-egress interface-type interface-number
undo mirroring-group group-id monitor-egress interface-type
interface-number
```

在接口视图下：

```
mirroring-group group-id monitor-egress
undo mirroring-group group-id monitor-egress
```

## 【缺省情况】

未为远程源镜像组配置出端口。

## 【视图】

系统视图

接口视图

## 【缺省用户角色】

network-admin

## 【参数】

*group-id*：表示镜像组的编号，该镜像组必须存在，取值范围为 1~4。

*interface-type interface-number*：表示出端口。其中，*interface-type* 为端口类型和端口编号。

## 【使用指导】

只能为远程源镜像组配置出端口，不能为本地镜像组和远程目的镜像组配置出端口。

不能在出端口上配置下列功能：生成树协议、802.1X、IGMP Snooping、静态 ARP 和 MAC 地址学习，否则会影响镜像功能的正常使用。

出端口不能是现有镜像组的成员端口。

出端口不能是聚合组的成员端口。

### 【举例】

# 创建远程源镜像组 1，并在系统视图下配置其出端口为 GigabitEthernet1/0/1。

```
<Sysname> system-view
[Sysname] mirroring-group 1 remote-source
[Sysname] mirroring-group 1 monitor-egress gigabitethernet 1/0/1
```

# 创建远程源镜像组 2，并在接口视图下配置其出端口为 GigabitEthernet1/0/2。

```
<Sysname> system-view
[Sysname] mirroring-group 2 remote-source
[Sysname] interface gigabitethernet 1/0/2
[Sysname-GigabitEthernet1/0/2] mirroring-group 2 monitor-egress
```

### 【相关命令】

- **mirroring-group**

## 1.1.6 mirroring-group monitor-port (interface view)

**mirroring-group monitor-port** 命令用来配置端口为镜像组的目的端口。

**undo mirroring-group monitor-port** 命令用来恢复缺省情况。

### 【命令】

```
mirroring-group group-id monitor-port
undo mirroring-group group-id monitor-port
```

### 【缺省情况】

未配置端口为镜像组的目的端口。

### 【视图】

接口视图

### 【缺省用户角色】

network-admin

### 【参数】

*group-id*: 镜像组。*group-id* 表示镜像组的编号，该镜像组必须存在，取值范围为 1~4。

### 【使用指导】

只能为本地镜像组或远程目的镜像组配置目的端口，不能为远程源镜像组配置目的端口。

请不要在目的端口上使能生成树协议，否则会影响镜像功能的正常使用。

当聚合接口作为目的端口时，请勿将其成员端口配置为源端口，否则会影响镜像功能的正常使用。

从目的端口发出的报文包括镜像报文和其他端口正常转发来的报文。为了保证数据监测设备只对镜像报文进行分析，请将目的端口只用于端口镜像，不作其他用途。

目的端口不能是现有镜像组的成员端口。

目的端口不能是聚合组的成员端口。

一个镜像组内只能配置一个目的端口。

### 【举例】

# 创建本地镜像组 1，并配置其目的端口为 GigabitEthernet1/0/1。

```

<Sysname> system-view
[Sysname] mirroring-group 1 local
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] mirroring-group 1 monitor-port
# 创建远程目的镜像组 2，并配置其目的端口为 GigabitEthernet1/0/2。
<Sysname> system-view
[Sysname] mirroring-group 2 remote-destination
[Sysname] interface gigabitethernet 1/0/2
[Sysname-GigabitEthernet1/0/2] mirroring-group 2 monitor-port

```

## 【相关命令】

- **mirroring-group**

### 1.1.7 mirroring-group monitor-port (system view)

**mirroring-group monitor-port** 命令用来为镜像组配置目的端口。

**undo mirroring-group monitor-port** 命令用来取消镜像组的目的端口配置。

## 【命令】

```

mirroring-group group-id monitor-port interface-type interface-number
undo mirroring-group group-id monitor-port interface-type interface-number

```

## 【缺省情况】

未为镜像组配置目的端口。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

*group-id*: 表示镜像组的编号，该镜像组必须存在，取值范围为 1~4。

*interface-type interface-number*: 表示目的端口。其中，*interface-type* 为端口类型和端口编号。

## 【使用指导】

只能为本地镜像组或远程目的镜像组配置目的端口，不能为远程源镜像组配置目的端口。

不能在目的端口上使能生成树协议，否则会影响镜像功能的正常使用。

当聚合接口作为目的端口时，请勿将其成员端口配置为源端口，否则会影响镜像功能的正常使用。

从目的端口发出的报文包括镜像报文和其他端口正常转发来的报文。为了保证数据监测设备只对镜像报文进行分析，请将目的端口只用于端口镜像，不作其他用途。

目的端口不能是现有镜像组的成员端口。

目的端口不能是聚合组的成员端口。

一个镜像组内只能配置一个目的端口。

### 【举例】

```
# 创建本地镜像组 1，并配置其目的端口为 GigabitEthernet1/0/1。
<Sysname> system-view
[Sysname] mirroring-group 1 local
[Sysname] mirroring-group 1 monitor-port gigabitethernet 1/0/1
# 创建远程目的镜像组 2，并配置其目的端口为 GigabitEthernet1/0/2。
<Sysname> system-view
[Sysname] mirroring-group 2 remote-destination
[Sysname] mirroring-group 2 monitor-port gigabitethernet 1/0/2
```

### 【相关命令】

- **mirroring-group**

#### 1.1.8 mirroring-group reflector-port

**mirroring-group reflector-port** 命令用来为远程源镜像组配置反射端口。

**undo mirroring-group reflector-port** 命令用来恢复缺省情况。

### 【命令】

在系统视图下：

```
mirroring-group group-id reflector-port interface-type interface-number
undo mirroring-group group-id reflector-port interface-type
interface-number
```

在接口视图下：

```
mirroring-group group-id reflector-port
undo mirroring-group group-id reflector-port
```

### 【缺省情况】

未为远程源镜像组配置反射端口。

### 【视图】

系统视图

接口视图

### 【缺省用户角色】

network-admin

### 【参数】

*group-id*：表示镜像组的编号，该镜像组必须存在，取值范围为 1~4。

*interface-type interface-number*：表示反射端口。其中，*interface-type* *interface-number* 为端口类型和端口编号。

### 【使用指导】

只能为远程源镜像组配置反射端口，不能为本地镜像组和远程目的镜像组配置反射端口。

建议选择设备上未被使用的端口作为反射端口，并不要在该端口上连接网线，否则会影响镜像功能的正常使用。

在将端口配置为反射端口时，该端口上已存在的所有配置都将被清除；在配置为反射端口后，该端口上不能再配置其他业务。

当 IRF 端口只绑定了一个物理端口时，请勿将该物理端口配置为反射端口，以免 IRF 分裂。

当端口已配置为反射端口后，不能再修改其双工模式、端口速率和 MDI 属性值，即这些属性只能取缺省值。

反射端口不能是聚合组的成员端口。

### 【举例】

# 创建远程源镜像组 1，并在系统视图下配置其反射端口为 GigabitEthernet1/0/1。

```
<Sysname> system-view
[Sysname] mirroring-group 1 remote-source
[Sysname] mirroring-group 1 reflector-port gigabitethernet 1/0/1
This operation may delete all settings made on the interface. Continue? [Y/N]: y
```

# 创建远程源镜像组 2，并在接口视图下配置其反射端口为 GigabitEthernet1/0/2。

```
<Sysname> system-view
[Sysname] mirroring-group 2 remote-source
[Sysname] interface gigabitethernet 1/0/2
[Sysname-GigabitEthernet1/0/2] mirroring-group 2 reflector-port
This operation may delete all settings made on the interface. Continue? [Y/N]: y
```

### 【相关命令】

- **mirroring-group**

## 1.1.9 mirroring-group remote-probe vlan

**mirroring-group remote-probe vlan** 命令用来为镜像组配置远程镜像 VLAN。

**undo mirroring-group remote-probe vlan** 命令用来恢复缺省情况。

### 【命令】

```
mirroring-group group-id remote-probe vlan vlan-id
undo mirroring-group group-id remote-probe vlan vlan-id
```

### 【缺省情况】

未为镜像组配置远程镜像 VLAN。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**group-id**: 表示镜像组的编号，该镜像组必须存在，取值范围为 1~4。

**vlan-id**: 表示远程镜像 VLAN 的编号。

### 【使用指导】

只能为远程源镜像组和远程目的镜像组配置远程镜像 VLAN，不能为本地镜像组配置远程镜像 VLAN。

当一个 VLAN 已被指定为远程镜像 VLAN 后，请不要将该 VLAN 再作其他用途。

源设备和目的设备上的远程镜像组必须使用相同的远程镜像 VLAN。

只能将已存在的静态 VLAN 配置为远程镜像 VLAN，且一个 VLAN 只能配置为一个镜像组的远程镜像 VLAN。

当某 VLAN 被配置为远程镜像 VLAN 后，必须先删除远程镜像 VLAN 的配置才能删除该 VLAN。

### 【举例】

# 创建远程源镜像组 1，并为其配置远程镜像 VLAN 为 VLAN 10。

```
<Sysname> system-view
[Sysname] mirroring-group 1 remote-source
[Sysname] mirroring-group 1 remote-probe vlan 10
```

# 创建远程目的镜像组 2，并为其配置远程镜像 VLAN 为 VLAN 20。

```
<Sysname> system-view
[Sysname] mirroring-group 2 remote-destination
[Sysname] mirroring-group 2 remote-probe vlan 20
```

### 【相关命令】

- **mirroring-group**



## 2 流镜像

### 2.1 流镜像配置命令

#### 2.1.1 mirror-to cpu

**mirror-to cpu** 命令用来配置流镜像到 CPU。

**undo mirror-to** 命令用来取消流镜像到 CPU。

##### 【命令】

```
mirror-to cpu
undo mirror-to cpu
```

##### 【缺省情况】

未配置流镜像到 CPU。

##### 【视图】

流行为视图

##### 【缺省用户角色】

network-admin

##### 【举例】

# 配置流行为 1，并在该流行为中配置流镜像到 CPU。

```
<Sysname> system-view
[Sysname] traffic behavior 1
[Sysname-behavior-1] mirror-to cpu
```

#### 2.1.2 mirror-to interface

**mirror-to interface** 命令用来配置流镜像到接口。

**undo mirror-to interface** 命令用来取消流镜像到接口。

##### 【命令】

```
mirror-to interface interface-type interface-number
undo mirror-to interface interface-type interface-number
```

##### 【缺省情况】

未配置流镜像到接口。

##### 【视图】

流行为视图

##### 【缺省用户角色】

network-admin

### 【参数】

*interface-type interface-number*: 表示流镜像接口的接口类型和接口编号。

### 【使用指导】

设备最多支持配置四个用于流镜像的流行为，如果多于四个流行为，则只有前四个流行为生效。  
同一流行为中只能配置一个目的接口，多次执行该命令，仅最后一次执行的命令生效。

### 【举例】

# 配置流行为 1，并在该流行为中配置流镜像到接口 GigabitEthernet1/0/1。

```
<Sysname> system-view
[Sysname] traffic behavior 1
[Sysname-behavior-1] mirror-to interface gigabitethernet 1/0/1
```

# 目 录

1 sFlow .....	1-1
1.1 sFlow配置命令.....	1-1
1.1.1 display sflow.....	1-1
1.1.2 sflow agent.....	1-2
1.1.3 sflow collector .....	1-3
1.1.4 sflow counter collector .....	1-4
1.1.5 sflow counter interval.....	1-4
1.1.6 sflow flow collector .....	1-5
1.1.7 sflow flow max-header .....	1-6
1.1.8 sflow sampling-mode.....	1-6
1.1.9 sflow sampling-rate .....	1-7
1.1.10 sflow source.....	1-8

# 1 sFlow

## 1.1 sFlow配置命令

### 1.1.1 display sflow

**display sflow** 命令用来显示 sFlow 的配置和运行信息。

**【命令】**

**display sflow**

**【视图】**

任意视图

**【缺省用户角色】**

network-admin  
network-operator

**【举例】**

```
# 显示 sFlow 的配置和运行信息。
<Sysname> display sflow
sFlow datagram version: 5
Global information:
Agent IP: 10.10.10.1(CLI)
Source address: 10.0.0.1 2001::1
Collector information:
ID      IP             Port  Aging      Size VPN-instance Description
1       22:2:20::10      6535  N/A        1400                netserver
2       192.168.3.5      6543  500        1400                Office
Port information:
Interface  CID  Interval(s) FID  MaxHLen Rate      Mode      Status
GE1/0/1    1    100         1    128    1000    Random   Active
GE1/0/2    2    100         2    128    1000    Random   Active
```

表1-1 display sflow 命令显示信息描述表

字段	描述
sFlow datagram version	sFlow报文版本号，取值只能为5，表示当前仅支持发送版本号为5的sFlow报文
Global information	sFlow全局信息
Agent IP	sFlow Agent的IP地址： <ul style="list-style-type: none"><li>CLI：表示手工配置的 IP 地址</li><li>Auto：表示自动查找到的 IP 地址</li></ul>
Source address	sFlow报文的源地址
Collector information	sFlow Collector信息

字段	描述
ID	sFlow Collector编号
IP	接收sFlow报文的sFlow Collector的IP地址
Port	接收sFlow报文的sFlow Collector的端口号
Aging	sFlow Collector的剩余存活时间。如果显示为N/A，则表示对应的sFlow Collector不会老化
Size	每次发送sFlow报文时，sFlow数据部分的最大长度
VPN-instance	（暂不支持）sFlow Collector的VPN实例名
Description	sFlow Collector的描述信息
Port information	已配置sFlow功能的接口信息
Interface	已配置sFlow功能的接口
CID	经过Counter采样后，sFlow Agent输出sFlow报文的目的地sFlow Collector编号。如果没有指定sFlow Collector编号，显示为0
Interval(s)	Counter采样的时间间隔
FID	经过Flow采样后，sFlow Agent输出sFlow报文的目的地sFlow Collector编号。如果没有指定sFlow Collector编号，显示为0
MaxHLen	从原始报文的头开始，允许拷贝的最大字节数
Rate	Flow采样的报文采样率
Mode	Flow采样的采样模式
Status	接口的sFlow功能的启用状态，其可能的取值如下： <ul style="list-style-type: none"> <li>• Suspended: 表示因接口处于 down 状态而挂起</li> <li>• Active: 表示因接口处于 up 状态而生效</li> </ul>

### 1.1.2 sflow agent

**sflow agent** 命令用来配置 sFlow Agent 的 IP 地址。

**undo sflow agent** 命令用来恢复缺省情况。

#### 【命令】

```
sflow agent { ip ipv4-address | ipv6 ipv6-address }
undo sflow agent { ip | ipv6 }
```

#### 【缺省情况】

未配置 sFlow Agent 的 IP 地址。设备会定期检查是否配置了 sFlow Agent 的 IP 地址，如果未配置，设备会自动查找一个 IPv4 地址作为 sFlow Agent 的 IP 地址。自动查找的 IP 地址信息不会保存在设备上。

#### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**ip** *ipv4-address*: sFlow Agent 的 IPv4 地址。

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

### 【使用指导】

建议用户手工配置 sFlow Agent 的 IP 地址。

多次执行本命令，最后一次执行的命令生效。

### 【举例】

# 配置 sFlow Agent 的 IP 地址为 10.10.10.1。

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

```
[Sysname] sflow agent ip 10.10.10.1
```

## 1.1.3 sflow collector

**sflow collector** 命令用来创建并配置 sFlow Collector。

**undo sflow collector** 命令用来删除指定的 sFlow Collector 信息。

### 【命令】

```
sflow collector collector-id { ip ipv4-address | ipv6 ipv6-address } [ port  
port-number ] [ datagram-size size ] [ time-out seconds ] [ description  
string ]
```

```
undo sflow collector collector-id
```

### 【缺省情况】

未配置 sFlow Collector 的相关信息。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**collector-id**: sFlow Collector 的编号，编号的取值范围为 1~10。

**ip** *ipv4-address*: sFlow Collector 的 IPv4 地址。

**ipv6** *ipv6-address*: sFlow Collector 的 IPv6 地址。

**port** *port-number*: sFlow Collector 的 UDP 端口号，取值范围为 1~65535，缺省值为 6343。

**datagram-size** *size*: 发送 sFlow 报文时，sFlow 数据部分的最大长度，取值范围为 200~3000，单位为字节，缺省值为 1400。

**time-out** *seconds*: 配置的 sFlow Collector 的参数老化时间，当到达老化时间时，所配置的 sFlow Collector 的参数将被删除。取值范围为 1~2147483647，单位为秒。缺省情况下，配置的 sFlow Collector 的参数不老化。

**description string**: sFlow Collector 的描述信息，为 1~127 个字符的字符串，区分大小写。缺省情况下，sFlow Collector 的描述信息为“CLI Collector”。

#### 【举例】

# 配置编号为 2 的 Collector，目的 IP 为 3.3.3.1，端口号保持缺省值，描述信息为“netserver”，老化时间为 1200 秒，sFlow 数据部分的最大长度为 1000 字节。

```
<Sysname> system-view
[Sysname] sflow collector 2 ip 3.3.3.1 description netserver time-out 1200 datagram-size 1000
```

### 1.1.4 sflow counter collector

**sflow counter collector** 命令用来配置经过 Counter 采样后，sFlow Agent 输出 sFlow 报文的目的 sFlow Collector 编号。

**undo sflow counter collector** 命令用来恢复缺省情况。

#### 【命令】

```
sflow counter collector collector-id
undo sflow counter collector
```

#### 【缺省情况】

Counter 采样和 sFlow Collector 没有绑定关系，即没有指定目的 sFlow Collector 编号。

#### 【视图】

二层以太网接口视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**collector-id**: sFlow Collector 的编号，编号的取值范围为 1~10。

#### 【举例】

# 在 GigabitEthernet1/0/1 上配置经过 Counter 采样后，sFlow Agent 输出 sFlow 报文的目的 sFlow Collector 编号为 2。

```
<Sysname> system-view
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] sflow counter collector 2
```

### 1.1.5 sflow counter interval

**sflow counter interval** 命令用来开启 Counter 采样功能并配置 Counter 采样的时间间隔。

**undo sflow counter interval** 命令用来在关闭 Counter 采用功能。

#### 【命令】

```
sflow counter interval interval
undo sflow counter interval
```

### 【缺省情况】

Counter 采样功能处于关闭状态。

### 【视图】

二层以太网接口视图

### 【缺省用户角色】

network-admin

### 【参数】

*interval*: Counter 采样的时间间隔，取值范围为 2~86400，单位为秒。

### 【举例】

# 在 GigabitEthernet1/0/1 上开启 Counter 采样功能并配置 Counter 采样的时间间隔为 120 秒。

```
<Sysname> system-view
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] sflow counter interval 120
```

## 1.1.6 sflow flow collector

**sflow flow collector** 命令用来配置经过 Flow 采样后，sFlow Agent 输出 sFlow 报文的目的 sFlow Collector 编号。

**undo sflow flow collector** 命令用来恢复缺省情况。

### 【命令】

```
sflow flow collector collector-id
undo sflow flow collector
```

### 【缺省情况】

Flow 采样和 sFlow Collector 没有绑定关系，即没有指定目的 sFlow Collector 编号。

### 【视图】

二层以太网接口视图

### 【缺省用户角色】

network-admin

### 【参数】

*collector-id*: sFlow Collector 的编号，编号的取值范围为 1~10。

### 【举例】

# 在 GigabitEthernet1/0/1 上配置经过 Flow 采样后，sFlow Agent 输出 sFlow 报文的目的 sFlow Collector 编号为 2。

```
<Sysname> system-view
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] sflow flow collector 2
```



### 1.1.7 sflow flow max-header

**sflow flow max-header** 命令用来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。

**undo sflow flow max-header** 命令用来恢复缺省情况。

#### 【命令】

```
sflow flow max-header length
undo sflow flow max-header
```

#### 【缺省情况】

从原始报文的头部开始，允许拷贝的最大字节数为 128 字节。

#### 【视图】

二层以太网接口视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**length**: 从原始报文的头部开始，允许拷贝的最大字节数，取值范围为 18~512。建议用户使用缺省配置。

#### 【举例】

# 在 GigabitEthernet1/0/1 上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为 60 字节。

```
<Sysname> system-view
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] sflow flow max-header 60
```

### 1.1.8 sflow sampling-mode

**sflow sampling-mode** 命令用来设置 Flow 采样的采样模式。

**undo sflow sampling-mode** 命令用来恢复缺省情况。

#### 【命令】

```
sflow sampling-mode random
undo sflow sampling-mode
```

#### 【视图】

二层以太网接口视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**random**: 表示采样模式为随机采样，采样率由 **sflow sampling-rate rate** 命令决定。设备会保持平均在每 *rate* 个报文中抽取一个报文进行采样，可能从每 *rate* 个报文中随机抽取任意一

个或多个报文进行采样，也可能在某段的 *rate* 个报文中不采样报文。例如，在配置此模式后，设定报文的采样率为 4000，设备可能会在 1~4000 个报文中选取其中的一个报文进行采样，在 4001~8000 个报文中选取其中的多个报文进行采样，在 8001~12000 个报文中不进行任何采样，但在长期时间内的总体趋势是 4000 个报文中抽取一个进行采样。

#### 【举例】

# 在 GigabitEthernet1/0/1 上配置 Flow 采样的采样模式为随机采样。

```
<Sysname> system-view
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] sflow sampling-mode random
```

#### 【相关命令】

- **sflow sampling-rate**

### 1.1.9 sflow sampling-rate

**sflow sampling-rate** 命令用来开启 Flow 采样功能并配置 Flow 采样的报文采样率，即在 *rate* 个报文中抽取一个报文进行采样。

**undo sflow sampling-rate** 命令用来关闭 Flow 采样功能。

#### 【命令】

```
sflow sampling-rate rate
undo sflow sampling-rate
```

#### 【缺省情况】

Flow 采样功能处于关闭状态

#### 【视图】

二层以太网接口视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**rate**: Flow 采样的报文采样率，取值范围为 1000~500000。建议用户配置报文采样率为 2 的 N 次方且大于等于 8192，例如 32768。

#### 【举例】

# 在 GigabitEthernet1/0/1 上开启 Flow 采样功能并配置 Flow 采样的报文采样率为 32768，即在 32768 个报文中抽取一个报文进行采样。

```
<Sysname> system-view
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] sflow sampling-rate 32768
```

#### 【相关命令】

- **sflow sampling-mode**

### 1.1.10 sflow source

**sflow source** 命令用来配置 sFlow 报文的源 IP 地址。

**undo sflow source** 命令用来恢复缺省情况。

#### 【命令】

```
sflow source { ip ipv4-address | ipv6 ipv6-address } *  
undo sflow source { ip | ipv6 } *
```

#### 【缺省情况】

设备使用路由决定的源 IP 地址作为 sFlow 报文的源 IP 地址。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**ip** *ipv4-address*: sFlow 报文的源 IPv4 地址。

**ipv6** *ipv6-address*: sFlow 报文的源 IPv6 地址。

#### 【举例】

# 配置 sFlow 报文的源 IPv4 地址为 10.0.0.1。

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

```
[Sysname] sflow source ip 10.0.0.1
```

# 目 录

1 信息中心.....	1-1
1.1 信息中心配置命令.....	1-1
1.1.1 diagnostic-logfile save .....	1-1
1.1.2 display diagnostic-logfile summary .....	1-1
1.1.3 display info-center.....	1-2
1.1.4 display logbuffer .....	1-4
1.1.5 display logbuffer summary.....	1-5
1.1.6 display logfile summary.....	1-6
1.1.7 display security-logfile summary.....	1-7
1.1.8 enable log updown.....	1-8
1.1.9 info-center diagnostic-logfile directory .....	1-8
1.1.10 info-center diagnostic-logfile enable .....	1-9
1.1.11 info-center diagnostic-logfile frequency .....	1-10
1.1.12 info-center diagnostic-logfile quota.....	1-10
1.1.13 info-center enable .....	1-11
1.1.14 info-center format.....	1-11
1.1.15 info-center logbuffer.....	1-12
1.1.16 info-center logbuffer size .....	1-12
1.1.17 info-center logfile directory .....	1-13
1.1.18 info-center logfile enable.....	1-14
1.1.19 info-center logfile frequency .....	1-14
1.1.20 info-center logfile overwrite-protection .....	1-15
1.1.21 info-center logfile size-quota .....	1-16
1.1.22 info-center logging suppress duplicates .....	1-16
1.1.23 info-center logging suppress module .....	1-17
1.1.24 info-center loghost.....	1-18
1.1.25 info-center loghost source.....	1-19
1.1.26 info-center security-logfile alarm-threshold.....	1-19
1.1.27 info-center security-logfile directory .....	1-20
1.1.28 info-center security-logfile enable .....	1-21
1.1.29 info-center security-logfile frequency .....	1-21
1.1.30 info-center security-logfile size-quota.....	1-22
1.1.31 info-center source .....	1-23

1.1.32 info-center synchronous .....	1-24
1.1.33 info-center syslog min-age .....	1-25
1.1.34 info-center syslog trap buffersize .....	1-26
1.1.35 info-center timestamp .....	1-26
1.1.36 info-center timestamp loghost .....	1-27
1.1.37 info-center trace-logfile quota .....	1-28
1.1.38 logfile save .....	1-29
1.1.39 reset logbuffer .....	1-29
1.1.40 security-logfile save .....	1-30
1.1.41 snmp-agent trap enable syslog .....	1-30
1.1.42 terminal debugging .....	1-31
1.1.43 terminal logging level .....	1-32
1.1.44 terminal monitor .....	1-33

# 1 信息中心

## 1.1 信息中心配置命令



说明

设备运行于 FIPS 模式时，本特性部分配置相对于非 FIPS 模式有所变化，具体差异请见本文相关描述。有关 FIPS 模式的详细介绍请参见“安全配置指导”中的“FIPS”。

### 1.1.1 diagnostic-logfile save

**diagnostic-logfile save** 命令用来手动将诊断日志文件缓冲区中的内容全部保存到诊断日志文件。

#### 【命令】

**diagnostic-logfile save**

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

#### 【使用指导】

诊断日志文件的保存路径可以通过 **info-center diagnostic-logfile directory** 命令配置。

诊断日志文件保存成功后，诊断日志文件缓冲区中的内容会被清空。

如果执行本命令时，诊断日志文件缓冲区中没有日志，本命令会执行成功，但不会创建或更新已有诊断日志文件。

#### 【举例】

# 手动将诊断日志文件缓冲区中的内容保存到诊断日志文件。

```
<Sysname> diagnostic-logfile save
```

```
The contents in the diagnostic log file buffer have been saved to the file  
flash:/diagfile/diagfile.log.
```

#### 【相关命令】

- **info-center diagnostic-logfile enable**
- **info-center diagnostic-logfile directory**

### 1.1.2 display diagnostic-logfile summary

**display diagnostic-logfile summary** 命令用来显示诊断日志文件的配置。

【命令】

**display diagnostic-logfile summary**

【视图】

任意视图

【缺省用户角色】

network-admin  
network-operator

【举例】

# 显示诊断日志文件配置。

```
<Sysname> display diagnostic-logfile summary
Diagnostic log file: Enabled.
Diagnostic log file size quota: 10 MB
Diagnostic log file directory: flash:/diagfile
Writing frequency: 24 hour 0 min 0 sec
```

表1-1 display logfile summary 命令显示信息描述表

字段	描述
Diagnostic log file	诊断日志文件当前的状态： <ul style="list-style-type: none"><li>• Enabled: 表示已开启</li><li>• Disabled: 表示未开启</li></ul>
Diagnostic log file size quota	单个诊断日志文件最大能占用的存储空间的大小，单位为MB
Diagnostic log file directory	诊断日志文件存储的路径
Writing frequency	系统自动保存诊断日志文件的频率

1.1.3 display info-center

**display info-center** 命令用来显示各个输出方向的信息。

【命令】

**display info-center**

【视图】

任意视图

【缺省用户角色】

network-admin  
network-operator

【举例】

# 显示各个输出方向的信息。

```
<Sysname> display info-center
Information Center: Enabled
```

```

Console: Enabled
Monitor: Enabled
Log host: Enabled
    192.168.0.1,
    port number: 5000, host facility: local7
Log buffer: Enabled
    Max buffer size 1024, current buffer size 512,
    Current messages 0, dropped messages 0, overwritten messages 0
Log file: Enabled
Security log file: Enabled
Information timestamp format:
    Log host: Date
    Other output destination: Date

```

表1-2 display info-center 命令显示信息描述表

字段	描述
Information Center	信息中心当前的状态： <ul style="list-style-type: none"> <li>Enabled: 表示已开启</li> <li>Disabled: 表示未开启</li> </ul>
Console	控制台方向当前的状态： <ul style="list-style-type: none"> <li>Enabled: 表示已开启</li> <li>Disabled: 表示未开启</li> </ul>
Monitor	监视终端方向当前的状态： <ul style="list-style-type: none"> <li>Enabled: 表示已开启</li> <li>Disabled: 表示未开启</li> </ul>
Log host: Enabled 192.168.0.1, port number: 5000, host facility: local7	日志主机方向的信息（只有通过 <b>info-center loghost</b> 命令配置后，才有下面具体的显示内容），包括日志主机的IP地址，日志主机接收日志信息的端口，日志主机的记录工具
Log buffer: Enabled Max buffer size 1024, current buffer size 512, Current messages 0, dropped messages 0, overwritten messages 0	日志缓冲区方向的信息，包括开启状态、最大容量、当前容量、当前信息数、已丢弃的信息数、被覆盖的信息数
Log file	日志文件方向当前的状态： <ul style="list-style-type: none"> <li>Enabled: 表示已开启</li> <li>Disabled: 表示未开启</li> </ul>
Security log file	安全日志文件方向当前的状态： <ul style="list-style-type: none"> <li>Enabled: 表示已开启</li> <li>Disabled: 表示未开启</li> </ul>
Information timestamp format: Loghost: Date Other output destination: Date	信息时间戳配置，包括日志主机输出方向和非日志主机输出方向日志信息的时间戳类型，分为boot、date、iso、none和no-year-date五种



## 1.1.4 display logbuffer

**display logbuffer** 命令用来显示日志缓冲区的状态和日志缓冲区记录的日志信息。

### 【命令】

```
display logbuffer [ reverse ] [ level severity | size buffersize | slot slot-number ] *
```

### 【视图】

任意视图

### 【缺省用户角色】

network-admin  
network-operator

### 【参数】

**reverse**: 指定日志的显示顺序为从新到旧。如果不指定该参数，将先显示旧日志，最后显示最新的日志。

**level severity**: 显示日志缓存中指定级别日志的信息，*severity* 表示信息级别，取值范围为 0~7。不带该参数时将显示系统日志缓冲区内所有级别的日志信息。

表1-3 信息级别列表

数值	信息级别	描述
0	emergency	表示设备不可用的信息，如系统授权已到期
1	alert	表示设备出现重大故障，需要立刻做出反应的信息，如流量超出接口上限
2	critical	表示严重信息，如设备温度已经超过预警值，设备电源、风扇出现故障等
3	error	表示错误信息，如接口链路状态变化等
4	warning	表示警告信息，如接口连接断开，内存耗尽告警等
5	notification	表示正常出现但是重要的信息，如通过终端登录设备，设备重启等
6	informational	表示需要记录的通知信息，如通过命令行输入命令的记录信息，执行ping命令的日志信息等
7	debugging	表示调试过程产生的信息

**size buffersize**: 显示日志缓冲区中指定条数的最新日志。*buffersize* 表示要显示的日志缓冲区中最新的日志信息的条数，取值范围为 1~1024。不带该参数时将显示系统日志缓冲区内所有的日志信息。

**slot slot-number**: 显示指定成员设备的日志缓冲区状态及日志缓冲区中记录的日志信息。*slot-number* 表示设备在 IRF 中的成员编号。如不指定该参数，将显示所有成员设备的日志缓冲区状态及日志缓冲区中记录的日志信息。

### 【举例】

# 显示系统日志缓冲区的状态和缓冲区记录的日志信息。

```
<Sysname> display logbuffer
```

```

Log buffer: Enabled
Max buffer size: 1024
Actual buffer size: 512
Dropped messages: 0
Overwritten messages: 718
Current messages: 512
%Jun 17 15:57:09:578 2016 Sysname SYSLOG/7/SYS_RESTART:System restarted --
其它显示信息略……。

```

表1-4 display logbuffer 命令显示信息描述表

字段	描述
Log buffer	是否允许输出到日志缓冲区方向： <ul style="list-style-type: none"> <li>Enabled: 表示允许</li> <li>Disabled: 表示不允许</li> </ul>
Max buffer size	允许的日志缓冲区可存储的最大信息条数
Actual buffer size	当前配置的日志缓冲区可存储的最大信息条数
Dropped messages	被丢弃的信息数（内存分配失败或分配日志缓冲区过小时丢失的信息数）
Overwritten messages	被覆盖的信息数（如果缓冲区存储空间不足，最早收到的信息数会被新的信息覆盖掉）
Current messages	当前记录的信息数

#### 【相关命令】

- info-center logbuffer
- reset logbuffer

### 1.1.5 display logbuffer summary

**display logbuffer summary** 命令用来显示系统日志缓冲区的概要信息。

#### 【命令】

**display logbuffer summary** [ **level severity** | **slot slot-number** ] \*

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin  
network-operator

#### 【参数】

**level severity**: 显示日志缓存中指定级别日志的概要信息，*severity*表示信息级别，取值范围为0~7，具体内容请参见 [表 1-3](#)。不带该参数时，将显示系统日志缓冲区内所有级别的日志概要信息。

**slot slot-number:**显示指定成员设备的日志缓冲区的概要信息。*slot-number* 表示设备在 IRF 中的成员编号。如不指定该参数，将显示所有成员设备上的日志缓冲区概要信息。

【举例】

```
# 显示系统日志缓冲区的概要信息。
<Sysname> display logbuffer summary
  Slot EMERG ALERT  CRIT ERROR  WARN NOTIF  INFO DEBUG
    1     0     0     0     7     0    34    38     0
```

表1-5 display logbuffer summary 命令显示信息描述表

字段	描述
EMERG	emergency级别的信息数，请参见 <a href="#">表1-3</a>
ALERT	alert级别的信息数，请参见 <a href="#">表1-3</a>
CRIT	critical级别的信息数，请参见 <a href="#">表1-3</a>
ERROR	error级别的信息数，请参见 <a href="#">表1-3</a>
WARN	warning级别的信息数，请参见 <a href="#">表1-3</a>
NOTIF	notification级别的信息数，请参见 <a href="#">表1-3</a>
INFO	informational级别的信息数，请参见 <a href="#">表1-3</a>
DEBUG	debugging级别的信息数，请参见 <a href="#">表1-3</a>

1.1.6 display logfile summary

**display logfile summary** 命令用来显示日志文件的配置。

【命令】

```
display logfile summary
```

【视图】

任意视图

【缺省用户角色】

network-admin  
network-operator

【举例】

```
# 显示日志文件配置。
<Sysname> display logfile summary
  Log file: Enabled.
  Log file size quota: 10 MB
  Log file directory: flash:/logfile
  Writing frequency: 24 hour 0 min 10 sec
```

表1-6 display logfile summary 命令显示信息描述表

字段	描述
Log file	是否允许输出到日志文件方向： <ul style="list-style-type: none"><li>• Enabled: 表示已开启</li><li>• Disabled: 表示未开启</li></ul>
Log file size quota	单个日志文件最大能占用的存储空间的大小，单位为MB
Log file directory	日志文件存储的路径
Writing frequency	系统自动保存日志文件的频率

1.1.7 display security-logfile summary

**display security-logfile summary** 命令用来显示安全日志文件的概要信息。

【命令】

**display security-logfile summary**

【视图】

任意视图

【缺省用户角色】

security-audit

【使用指导】

只有具有安全日志管理员角色的本地用户才能使用本命令。安全日志管理员的配置请参见“安全命令参考”中的“AAA”。

【举例】

```
# 显示安全日志文件概要信息。
<Sysname> display security-logfile summary
Security log file: Enabled
Security log file size quota: 10 MB
Security log file directory: flash:/seclog
Alarm threshold: 80%
Current usage: 30%
Writing frequency: 24 hour 0 min 0 sec
```

表1-7 display security-logfile summary 命令显示信息描述表

字段	描述
Security log file	安全日志文件当前的状态： <ul style="list-style-type: none"><li>• Enabled: 表示已开启</li><li>• Disabled: 表示未开启</li></ul>
Security log file size quota	单个安全日志文件最大能占用的存储空间的大小
Security log file directory	安全日志文件存储的路径

字段	描述
Alarm-threshold	安全日志文件使用率告警门限
Current usage	当前的安全日志文件使用率
Writing frequency	系统自动保存安全日志文件的频率

#### 【相关命令】

- **authorization-attribute**（安全命令参考/AAA）

### 1.1.8 enable log updown

**enable log updown** 命令用来允许端口在状态发生改变时生成 Link up 和 Link down 的日志信息。

**undo enable log updown** 命令用来禁止端口在状态发生改变时生成 Link up 和 Link down 的日志信息。

#### 【命令】

```
enable log updown
undo enable log updown
```

#### 【缺省情况】

允许所有端口在状态发生改变时生成端口 Link up 和 Link down 的日志信息。

#### 【视图】

接口视图

#### 【缺省用户角色】

network-admin

#### 【举例】

```
# 禁止端口 GigabitEthernet1/0/1 在状态发生改变时生成 Link up 和 Link down 的日志信息。
<Sysname> system-view
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] undo enable log updown
```

### 1.1.9 info-center diagnostic-logfile directory

**info-center diagnostic-logfile directory** 命令用来修改存储诊断日志文件的路径。

#### 【命令】

```
info-center diagnostic-logfile directory dir-name
```

#### 【缺省情况】

存储诊断日志文件路径为 flash:/diagfile。

#### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*dir-name*: 诊断日志文件存储的路径，为 1~64 个字符的字符串。

### 【使用指导】

在执行该配置前，存储诊断日志文件的文件夹必须为当前已创建的目录。

配置会在 IRF 重启或主从设备倒换后失效。

### 【举例】

```
# 配置存放诊断日志文件的目录为 flash:/test。
<Sysname> mkdir test
Creating directory flash:/test... Done.
<Sysname> system-view
[Sysname] info-center diagnostic-logfile directory flash:/test
```

## 1.1.10 info-center diagnostic-logfile enable

**info-center diagnostic-logfile enable** 命令用来开启诊断日志同步保存功能。

**undo info-center diagose-logfile enable** 命令用来关闭诊断日志同步保存功能。

### 【命令】

```
info-center diagnostic-logfile enable
undo info-center diagnostic-logfile enable
```

### 【缺省情况】

诊断日志同步保存功能处于开启状态。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【使用指导】

开启诊断日志同步保存功能后，系统将诊断日志进行集中处理：当生成的诊断日志时，系统会将诊断日志信息同步保存到诊断日志文件。这样既实现了诊断日志的集中管理，又有利于用户随时快捷地查看诊断日志，了解设备状态。

### 【举例】

```
# 开启诊断日志同步保存功能。
<Sysname> system-view
[Sysname] info-center diagnostic-logfile enable
```

### 1.1.11 info-center diagnostic-logfile frequency

**info-center diagnostic-logfile frequency** 命令用来配置设备自动保存诊断日志文件的频率。

**undo info-center diagnostic-logfile frequency** 命令用来恢复缺省情况。

#### 【命令】

**info-center diagnostic-logfile frequency** *freq-sec*

**undo info-center diagnostic-logfile frequency**

#### 【缺省情况】

设备自动保存诊断日志文件的频率为 86400 秒。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*freq-sec*: 系统自动保存诊断日志文件的频率，取值范围为 10~86400，单位为秒。

#### 【使用指导】

配置设备自动保存诊断日志文件的频率后，诊断日志会先被输出到诊断日志文件缓冲区（**diagnostic-logfile buffer**），系统会按照配置中指定的频率将诊断日志文件缓冲区的内容写入诊断日志文件。

#### 【举例】

# 配置诊断日志自动保存到文件的频率为 600 秒。

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

```
[Sysname] info-center diagnostic-logfile frequency 600
```

#### 【相关命令】

- **info-center diagnostic-logfile enable**

### 1.1.12 info-center diagnostic-logfile quota

**info-center diagnostic-logfile quota** 命令用来配置单个诊断日志文件最大可占用的存储空间的大小。

**undo info-center diagnostic-logfile quota** 命令用来恢复缺省情况。

#### 【命令】

**info-center diagnostic-logfile quota** *size*

**undo info-center diagnostic-logfile quota**

#### 【缺省情况】

单个诊断日志文件最大能占用的存储空间为 10MB。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**size**: 单个诊断日志文件可使用的存储空间的最大值，取值范围为 1~10，单位为 MB。

### 【举例】

# 配置单个诊断日志文件最大能占用的存储空间的大小为 6MB。

```
<Sysname> system-view
[Sysname] info-center diagnostic-logfile quota 6
```

## 1.1.13 info-center enable

**info-center enable** 命令用来开启信息中心功能。

**undo info-center enable** 命令用来关闭信息中心功能。

### 【命令】

```
info-center enable
undo info-center enable
```

### 【缺省情况】

信息中心处于开启状态。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【举例】

# 开启信息中心功能。

```
<Sysname> system-view
[Sysname] info-center enable
Information center is enabled.
```

## 1.1.14 info-center format

**info-center format** 命令用来配置发往日志主机的日志信息的输出格式。

**undo info-center format** 命令用来恢复缺省情况。

### 【命令】

```
info-center format { cmcc | unicom }
undo info-center format
```

### 【缺省情况】

发往日志主机的日志信息的格式为非定制格式。



### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**cmcc:** 配置发往日志主机的日志信息的输出格式为中国移动格式。

**unicom:** 配置发往日志主机的日志信息的输出格式为中国联通格式。

### 【使用指导】

发往日志主机的日志信息有三种格式：非定制格式、中国联通格式和中国移动格式。有关日志信息格式的详细介绍请参见“网络管理和监控配置指导”中的“信息中心”。

### 【举例】

# 配置发往日志主机的日志信息的格式为中国联通格式。

```
<Sysname> system-view
[Sysname] info-center format unicom
```

## 1.1.15 info-center logbuffer

**info-center logbuffer** 命令用来允许日志信息输出到日志缓冲区。

**undo info-center logbuffer** 命令用来禁止日志信息输出到日志缓冲区。

### 【命令】

```
info-center logbuffer
undo info-center logbuffer
```

### 【缺省情况】

允许日志信息输出到日志缓冲区。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【举例】

# 配置允许日志信息输出到日志缓冲区。

```
<Sysname> system-view
[Sysname] info-center logbuffer
```

### 【相关命令】

- **display logbuffer**
- **info-center enable**

## 1.1.16 info-center logbuffer size

**info-center logbuffer size** 命令用来配置日志缓冲区可存储的信息条数。

**undo info-center logbuffer size** 命令用来恢复缺省情况。

**【命令】**

```
info-center logbuffer size buffersize  
undo info-center logbuffer size
```

**【缺省情况】**

日志缓冲区可存储 512 条信息。

**【视图】**

系统视图

**【缺省用户角色】**

network-admin

**【参数】**

*buffersize*: 日志缓冲区可存储的信息条数，取值范围为 0～1024，缺省值为 512。

**【举例】**

```
# 配置日志缓冲区可存储的信息条数为 50。  
<Sysname> system-view  
[Sysname] info-center logbuffer size 50  
# 恢复日志缓冲区可存储的信息条数为 512。  
<Sysname> system-view  
[Sysname] undo info-center logbuffer size
```

**【相关命令】**

- **display logbuffer**
- **info-center enable**

### 1.1.17 info-center logfile directory

**info-center logfile directory** 命令用来配置存放日志文件的目录。

**【命令】**

```
info-center logfile directory dir-name
```

**【缺省情况】**

存放日志文件的目录为 flash:/logfile。

**【视图】**

系统视图

**【缺省用户角色】**

network-admin

**【参数】**

*dir-name*: 存放日志文件的路径，为 1～64 个字符的字符串。

### 【使用指导】

在执行该配置前，存放日志文件的目录必须为当前已创建的目录。  
生成的日志文件的后缀名为.log。  
配置会在 IRF 重启或主从设备倒换后失效。

### 【举例】

```
# 配置存放日志文件的目录为 flash:/test.  
<Sysname> mkdir test  
Creating directory flash:/test... Done.  
<Sysname> system-view  
[Sysname] info-center logfile directory flash:/test
```

### 【相关命令】

- **info-center logfile enable**

#### 1.1.18 info-center logfile enable

**info-center logfile enable** 命令用来允许日志信息输出到日志文件。  
**undo info-center logfile enable** 命令用来禁止日志信息输出到日志文件。

### 【命令】

```
info-center logfile enable  
undo info-center logfile enable
```

### 【缺省情况】

允许日志信息输出到日志文件。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【举例】

```
# 配置允许日志信息输出到日志文件。  
<Sysname> system-view  
[Sysname] info-center logfile enable
```

#### 1.1.19 info-center logfile frequency

**info-center logfile frequency** 命令用来配置系统自动保存日志文件的频率。  
**undo info-center logfile frequency** 命令用来恢复缺省情况。

### 【命令】

```
info-center logfile frequency freq-sec  
undo info-center logfile frequency
```

### 【缺省情况】

设备自动保存日志文件的频率为 86400 秒。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*freq-sec*: 系统自动保存日志文件的频率，取值范围为 1～86400，单位为秒。

### 【使用指导】

配置系统自动保存日志文件的频率后，系统会按照指定的频率将日志文件缓冲区的内容写入日志文件。

### 【举例】

```
# 配置设备自动保存日志文件的频率为 60000 秒。  
<Sysname> system-view  
[Sysname] info-center logfile frequency 60000
```

### 【相关命令】

- **info-center logfile enable**

## 1.1.20 info-center logfile overwrite-protection

**info-center logfile overwrite-protection** 命令用来开启日志文件的写满保护功能。

**undo info-center logfile overwrite-protection** 命令用来关闭日志文件的写满保护功能。

### 【命令】

```
info-center logfile overwrite-protection [ all-port-powerdown ]  
undo info-center logfile overwrite-protection
```

### 【缺省情况】

日志文件的写满保护功能处于关闭状态。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**all-port-powerdown**: 表示如果日志文件已经达到限额或空间不足，则关闭所有的业务接口。  
如果不指定该参数，则表示当日志文件已经达到限额或空间不足时，不关闭所有的业务接口。

### 【使用指导】

本命令仅 FIPS 模式下支持。

如果日志文件的写满保护功能处于开启状态，则日志将从当前日志文件的尾部开始进行记录。在记录日志的过程中，如果日志文件的个数达到设备支持的最大值或者设备可用存储介质的空间不足，不再覆盖旧日志或删除最旧的日志文件，而是停止记录日志文件。

#### 【举例】

# 开启日志文件的写满保护功能。

```
<Sysname> system-view
[Sysname] info-center logfile overwrite-protection
```

### 1.1.21 info-center logfile size-quota

**info-center logfile size-quota** 命令用来配置单个日志文件最大能占用的存储空间的大小。

**undo info-center logfile size-quota** 命令用来恢复缺省情况。

#### 【命令】

```
info-center logfile size-quota size
undo info-center logfile size-quota
```

#### 【缺省情况】

单个日志文件最大能占用的存储空间为 10MB。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**size**: 单个日志文件可使用的存储空间的最大值，取值范围为 1~10，单位为 MB。

#### 【举例】

# 配置单个日志文件最大能占用的存储空间的大小为 6MB。

```
<Sysname> system-view
[Sysname] info-center logfile size-quota 6
```

#### 【相关命令】

- **info-center logfile enable**

### 1.1.22 info-center logging suppress duplicates

**info-center logging suppress duplicates** 命令用来开启抑制重复日志输出功能。

**undo info-center logging suppress duplicate** 命令用关闭抑制重复日志输出功能。

#### 【命令】

```
info-center logging suppress duplicates
undo info-center logging suppress duplicates
```

#### 【缺省情况】

抑制重复日志输出功能处于关闭状态。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【使用指导】

当设备持续生成相同的日志，大量重复的信息会浪费设备资源和网络资源，并导致有用的信息被淹没，不利于设备的维护。为了避免此问题，可开启重复日志抑制功能。

## 【举例】

# 开启抑制重复日志输出功能。

```
<Sysname> system-view
[Sysname] info-center logging suppress duplicates
```

### 1.1.23 info-center logging suppress module

**info-center logging suppress module** 命令用来禁止指定模块日志的输出。

**undo info-center logging suppress module** 命令用来取消对指定模块日志输出的限制。

## 【命令】

```
info-center logging suppress module module-name mnemonic { all |
mnemonic-value }
undo info-center logging suppress module module-name mnemonic { all |
mnemonic-value }
```

## 【缺省情况】

未禁止指定模块日志的输出。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**module-name**: 指定需要被过滤的应用模块的名称，为 1~8 个字符的字符串，不区分大小写。执行 **info-center logging suppress module ?** 命令，设备会自动显示可配置的模块名称。

**mnemonic { all | mnemonic-value }**: 表示需要被过滤的应用模块的助记符。其中：

- **all**: 表示过滤该模块的所有日志。
- **mnemonic-value**: 表示助记符的全部内容，为 1~32 个字符的字符串，不区分大小写。如果输入的是助记符的部分内容，将不会禁止输出该助记符对应的日志。

## 【使用指导】

当系统日志较多时，对于用户不关心的日志，可以使用该命令，禁止指定模块日志的输出，或者禁止指定模块特定助记符的日志输出。

### 【举例】

```
# 禁止 SHELL 模块助记符为 SHELL_LOGIN 的日志输出。
<Sysname> system-view
[Sysname] info-center logging suppress module shell mnemonic shell_login
```

### 【相关命令】

- **info-center source**

## 1.1.24 info-center loghost

**info-center loghost** 命令用来配置日志主机及相关参数。

**undo info-center loghost** 命令用来取消日志主机及相关参数的配置。

### 【命令】

```
info-center loghost { hostname | ipv4-address | ipv6 ipv6-address } [ port
port-number ] [ dscp dscp-value ] [ facility local-number ]
undo info-center loghost { hostname | ipv4-address | ipv6 ipv6-address }
```

### 【缺省情况】

未配置日志主机及相关参数。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**hostname**: 指定日志的主机名，为 1~253 个字符的字符串，不区分大小写，字符串中可以包含字母、数字、“-”、“\_”和“.”。

**ipv4-address**: 指定日志主机的 IPv4 地址。

**ipv6 ipv6-address**: 指定日志主机的 IPv6 地址。

**port port-number**: 指定日志主机接收日志信息的端口号，取值范围为 1~65535，缺省值为 514。该参数的值需要和日志主机侧的配置一致，否则日志主机接收不到日志信息。

**dscp dscp-value**: 指定日志消息的 DSCP（Differentiated Services Code Point，差分服务代码点）优先级，取值范围为 0~63，缺省值为 0。值越大表示优先级越高。DSCP 优先级封装在 IP 报文中的 ToS 字段，用来表示报文自身的优先级，设备可根据该优先级决定报文传输的优先程度。

**facility local-number**: 配置日志主机的记录工具。取值范围为 local0~local7，缺省值为 local7。主要用于在日志主机端标志不同的日志来源，查找、过滤对应日志源的日志。

### 【使用指导】

用户只有使用 **info-center enable** 命令开启了信息中心功能，配置 **info-center loghost** 命令才会生效。

### 【举例】

```
# 配置系统向 IP 地址为 1.1.1.1 的日志主机发送日志信息。
```

```
<Sysname> system-view
[Sysname] info-center loghost 1.1.1.1
```

### 1.1.25 info-center loghost source

**info-center loghost source** 命令用来配置发送的日志信息的源 IP 地址。

**undo info-center loghost source** 命令用来恢复缺省情况。

#### 【命令】

```
info-center loghost source interface-type interface-number
undo info-center loghost source
```

#### 【缺省情况】

使用出接口的主 IP 地址作为发送的日志信息的源 IP 地址。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*interface-type interface-number*: 指定发送日志信息的源 IP 地址对应的出接口的类型和编号。

#### 【使用指导】

配置日志信息的源 IP 地址后，不管实际使用哪个物理接口发送日志信息，日志信息的源 IP 地址均为指定接口的主 IP 地址。

用户只有使用 **info-center enable** 命令开启了信息中心功能，配置 **info-center loghost source** 命令才会生效。

#### 【举例】

# 配置使用 Loopback0 的 IP 地址作为发送的日志信息的源 IP 地址。

```
<Sysname> system-view
[Sysname] interface loopback 0
[Sysname-LoopBack0] ip address 2.2.2.2 32
[Sysname-LoopBack0] quit
[Sysname] info-center loghost source loopback 0
```

### 1.1.26 info-center security-logfile alarm-threshold

**info-center security-logfile alarm-threshold** 命令用来配置安全日志文件使用率的告警门限。

**undo info-center security-logfile alarm-threshold** 命令用来恢复缺省情况。

#### 【命令】

```
info-center security-logfile alarm-threshold usage
undo info-center security-logfile alarm-threshold
```



### 【缺省情况】

安全日志文件使用率的告警门限是 80（即当安全日志文件使用率达到 80%时，系统会提醒用户）。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**usage**：安全日志文件使用率的告警门限，取值范围为 1～100 的整数。

### 【使用指导】

当安全日志文件大小达到上限时，新的安全日志会覆盖旧的安全日志，从而导致安全日志丢失。为防止这种情况发生，用户可以使用本命令配置安全日志文件使用率的告警门限。当使用率超过此门限值时，系统会发出日志提醒用户，此时，用户可以将安全日志文件进行备份，以防重要历史数据丢失。

### 【举例】

```
# 配置当安全日志文件使用率达到 90%时进行告警。
<Sysname> system-view
[Sysname] info-center security-logfile alarm-threshold 90
```

### 【相关命令】

- **info-center security-logfile size-quota**

## 1.1.27 info-center security-logfile directory

**info-center security-logfile directory** 命令用来修改存储安全日志文件的路径。

### 【命令】

**info-center security-logfile directory** *dir-name*

### 【缺省情况】

存储安全日志文件路径为 flash:/seclog。

### 【视图】

系统视图

### 【缺省用户角色】

security-audit

### 【参数】

**dir-name**：安全日志文件存储的路径，为 1～64 个字符的字符串。

### 【使用指导】

在执行该配置前，存储安全日志文件的文件夹必须为当前已创建的目录。

只有配置了安全日志管理员权限的本地用户才能使用本命令。安全日志管理员的配置请参见“安全命令参考”中的“AAA”。

配置会在 IRF 重启或主从设备倒换后失效。

#### 【举例】

```
# 配置存放安全日志文件的目录为 flash:/test。
<Sysname> mkdir test
Creating directory flash:/test... Done.
<Sysname> system-view
[Sysname] info-center security-logfile directory flash:/test
```

### 1.1.28 info-center security-logfile enable

**info-center security-logfile enable** 命令用来开启安全日志同步保存功能。

**undo info-center security-logfile enable** 命令用来关闭安全日志同步保存功能。

#### 【命令】

```
info-center security-logfile enable
undo info-center security-logfile enable
```

#### 【缺省情况】

安全日志同步保存功能处于关闭状态。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【使用指导】

开启安全日志同步保存功能后，系统将安全日志进行集中处理：当生成的日志信息中有安全日志，在不影响日志信息现有输出规则的前提下，系统会将安全日志信息同步保存到专用的安全日志文件。这样既实现了安全日志的集中管理，又有利于用户随时快捷地查看安全日志，了解设备状态。

#### 【举例】

```
# 开启安全日志同步保存功能。
<Sysname> system-view
[Sysname] info-center security-logfile enable
```

### 1.1.29 info-center security-logfile frequency

**info-center security-logfile frequency** 命令用来配置设备自动保存安全日志文件的频率。

**undo info-center security-logfile frequency** 命令用来恢复缺省情况。

#### 【命令】

```
info-center security-logfile frequency freq-sec
undo info-center security-logfile frequency
```

#### 【缺省情况】

设备自动保存安全日志文件的频率为 86400 秒。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

*freq-sec*: 系统自动保存安全日志文件的频率，取值范围为 10～86400，单位为秒。

## 【使用指导】

配置设备自动保存安全日志文件的频率后，安全日志会先被输出到安全日志文件缓冲区（*security-logfile buffer*），系统会按照配置中指定的频率将安全日志文件缓冲区的内容写入安全日志文件。

## 【举例】

# 配置安全日志自动保存到文件的频率为 600 秒。

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

```
[Sysname] info-center security-logfile frequency 600
```

## 【相关命令】

- **info-center security-logfile enable**

### 1.1.30 info-center security-logfile size-quota

**info-center security-logfile size-quota** 命令用来配置单个安全日志文件最大能占用的存储空间的大小。

**undo info-center security-logfile size-quota** 命令用来恢复缺省情况。

## 【命令】

```
info-center security-logfile size-quota size
```

```
undo info-center security-logfile size-quota
```

## 【缺省情况】

单个安全日志文件最大能占用的存储空间的大小为 10MB。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

*size*: 单个安全日志文件可使用的存储空间的最大值，取值范围为 1～10，单位为 MB。

## 【举例】

# 配置单个安全日志文件最大能占用的存储空间的大小为 6MB。

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

```
[Sysname] info-center security-logfile size-quota 6
```

## 【相关命令】

- `info-center security-logfile alarm-threshold`

### 1.1.31 info-center source

`info-center source` 命令用来配置日志信息的输出规则。

`undo info-center source` 命令用来恢复缺省情况。

## 【命令】

```
info-center source { module-name | default } { console | logbuffer | logfile |  
loghost | monitor } { deny | level severity }
```

```
undo info-center source { module-name | default } { console | logbuffer |  
logfile | loghost | monitor }
```

## 【缺省情况】

日志信息的输出规则请参见 [表 1-8](#)：

表1-8 输出方向的缺省输出规则

输出方向	允许输出的模块	log	security	diagnostic	hide
控制台	所有支持的模块	debugging	不能输出到控制台	不能输出到控制台	不能输出到控制台
监视终端	所有支持的模块	debugging	不能输出到控制台	不能输出到控制台	不能输出到控制台
日志主机	所有支持的模块	informational	不能输出到日志主机	不能输出到日志主机	informational
日志缓冲区	所有支持的模块	informational	不能输出到日志缓冲区	不能输出到日志缓冲区	informational
日志文件	所有支持的模块	informational	不能输出到日志文件	不能输出到日志文件	informational
安全日志文件	所有支持的模块，不能过滤	不能输出到安全日志文件	debugging（不能过滤）	不能输出到安全日志文件	不能输出到安全日志文件
诊断日志文件	所有支持的模块，不能过滤	不能输出到诊断日志文件	不能输出到诊断日志文件	debugging（不能过滤）	不能输出到诊断日志文件

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**module-name**：配置指定应用模块日志信息的输出规则。系统支持的来源模块可以通过在系统视图下输入 `info-center source ?` 进行查看。

**default**：配置所有应用模块日志信息的缺省输出规则。

**console**: 输出到控制台。

**logbuffer**: 输出到日志缓冲区。

**logfile**: 输出到日志文件。

**loghost**: 输出到日志主机。

**monitor**: 输出到监视终端。

**deny**: 禁止输出信息。

**level severity**: 指定信息级别，取值范围为 0~7，具体内容请参见 [表 1-3](#)。通过该参数可以控制允许/禁止输出的日志信息的最低级别。

### 【使用指导】

如果没有使用 *module-name* 参数为应用模块单独配置输出规则，则该模块使用 **default** 参数配置的输出规则。如果未使用 **default** 参数配置输出规则，则使用缺省输出规则。

给应用模块单独配置输出规则后，必须使用 *module-name* 参数来修改或删除该规则，使用 **default** 参数进行的新配置对该模块不生效。

多次使用 *module-name* 参数为同一应用模块配置输出规则，最后一次执行的命令生效；多次使用 **default** 参数配置缺省输出规则，最后一次执行的命令生效。

### 【举例】

# 只允许 VLAN 模块的信息输出控制台，且输出信息的级别为 **emergency** 的日志信息。

```
<Sysname> system-view
[Sysname] info-center source default console deny
[Sysname] info-center source vlan console level emergency
```

# 在前面例子基础上撤销控制台方向上 VLAN 模块的输出（即全部信息不在控制台显示）。

```
<Sysname> system-view
[Sysname] undo info-center source vlan console
```

## 1.1.32 info-center synchronous

**info-center synchronous** 命令用来开启命令行输入回显功能。

**undo info-center synchronous** 命令用来关闭命令行输入回显功能。

### 【命令】

```
info-center synchronous
undo info-center synchronous
```

### 【缺省情况】

命令行输入回显功能处于关闭状态。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

## 【使用指导】

当用户进行命令行、参数或者 Y/N 确认信息输入时，如果被大量的日志信息打断，用户可能记不清已经输入了哪些字符串，还需要输入哪些字符串。使用命令行输入回显功能，能够协助用户配置。系统会在日志信息输出完毕后回显用户已有的输入或者 Y/N 确认信息，以便用户继续执行配置。

## 【举例】

# 开启设备命令行输入回显功能，输入 **display current-configuration** 命令查看设备当前配置。

```
<Sysname> system-view
[Sysname] info-center synchronous
Info-center synchronous output is on
[Sysname] display current-
```

此时，收到日志报文，则系统在所有的日志报文显示完以后，附加显示用户的输入（此例为“display current-”）。

```
%May 21 14:33:19:425 2007 Sysname SHELL/4/LOGIN: VTY login from 192.168.1.44
[Sysname] display current-
```

此时，用户可以继续输入 **configuration**（完成命令 **display current-configuration** 的完整输入），回车，即可执行该命令。

# 开启设备命令行输入回显功能，保存当前配置（输入交互信息）。

```
<Sysname> system-view
[Sysname] info-center synchronous
Info-center synchronous output is on
[Sysname] save
The current configuration will be written to the device. Are you sure? [Y/N]:
```

此时，收到日志信息，则系统会在所有的日志报文显示完以后，附加显示[Y/N]：

```
%May 21 14:33:19:425 2007 Sysname SHELL/4/LOGIN: VTY login from 192.168.1.44
[Y/N]:
```

此时，用户可以输入 Y 或者 N，继续日志信息输出前的操作。

### 1.1.33 info-center syslog min-age

**info-center syslog min-age** 命令用来配置日志信息的最短保存时间。

**undo info-center syslog min-age** 命令用来恢复缺省情况。

## 【命令】

```
info-center syslog min-age min-age
undo info-center syslog min-age
```

## 【缺省情况】

未配置日志信息的最短保存时间。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

### 【参数】

*min-age*: 表示日志信息的最短保存时间，取值范围为 1～8760，单位为小时。

### 【举例】

# 将日志信息的最短保存时间配置为 168 小时。

```
<Sysname> system-view
[Sysname] info-center syslog min-age 168
```

## 1.1.34 info-center syslog trap buffersize

**info-center syslog trap** 命令用来配置日志告警缓冲区的大小。

**undo info-center syslog trap** 命令用来恢复缺省情况。

### 【命令】

```
info-center syslog trap buffersize buffersize
undo info-center syslog trap buffersize
```

### 【缺省情况】

日志告警缓冲区的大小为 1024 条。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*buffersize*: 表示日志告警缓冲区的大小，取值范围为 0～65535，单位为条。当取值为 0 时，表示不缓存日志告警。

### 【使用指导】

配置 **snmp-agent trap enable syslog** 命令后，设备会将日志封装成告警信息，存储在日志告警缓冲区。用户通过访问“日志告警缓冲区对应的 MIB 节点”，就可以直接读取日志告警。

通常情况下，日志告警缓冲区的大小采用缺省值即可。当日志告警缓冲区内存储的信息达到上限，且还需要存储信息时，新信息会覆盖旧信息。

### 【举例】

# 配置日志告警缓冲区的大小为 2048 条。

```
<Sysname> system-view
[Sysname] info-center syslog trap buffersize 2048
```

### 【相关命令】

- **snmp-agent trap enable syslog**

## 1.1.35 info-center timestamp

**info-center timestamp** 命令用来配置发往控制台、监视终端、日志缓冲区和日志文件方向的日志信息的时间戳输出格式。

**undo info-center timestamp** 命令用来恢复缺省情况。

#### 【命令】

```
info-center timestamp { boot | date | none }  
undo info-center timestamp
```

#### 【缺省情况】

时间戳输出格式为 **date** 格式。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**boot**: 系统启动后经历的时间，格式为：xxx.yyy，其中 xxx 是系统自启动后经历时间的毫秒数高 32 位，yyy 是低 32 位，形如 0.21990989（等效于 Jun 25 14:09:26:881 2007）。

**date**: 系统当前的日期和时间，格式为“MMM DD hh:mm:ss:xxx YYYY”，形如 Dec 8 10:12:21:708 2007。

- “MMM” 为英语月份的缩写，具体取值如下：Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec。
- “DD” 表示日期，如果日期的值小于 10，则格式为“空格+日期”，如“7”。
- “hh:mm:ss:xxx” 表示本地时间，hh 的取值范围为 00~23，mm 和 ss 的取值范围均为 00~59，xxx 的取值范围为 0~999。
- “YYYY” 表示年份。

**none**: 不带时间信息。

#### 【举例】

```
# 配置日志信息时间戳输出格式为 boot 格式。  
<Sysname> system-view  
[Sysname] info-center timestamp boot
```

#### 【相关命令】

- **info-center timestamp loghost**

### 1.1.36 info-center timestamp loghost

**info-center timestamp loghost** 命令用来配置发往日志主机的日志信息的时间戳输出格式。

**undo info-center timestamp loghost** 命令用来恢复缺省情况。

#### 【命令】

```
info-center timestamp loghost { date | iso [ with-timezone ] | no-year-date |  
none }  
undo info-center timestamp loghost
```



### 【缺省情况】

发往日志主机的日志信息的时间戳输出格式为 **date** 格式。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**date**: 系统当前日期和时间，格式为：mmm dd hh:mm:ss yyyy，形如 Dec 8 10:12:21 2007，但最终显示格式由日志主机决定。

**iso**: 配置时间戳采用 ISO 8601 标准格式，形如：2009-09-21T15:32:55。

**with-timezone**: 在时间戳中携带时区信息，形如：2009-09-21T15:32:55+01:00。不指定该参数时，表示在时间戳中不带时区信息。

**no-year-date**: 系统当前日期和时间，但不包含年份信息。

**none**: 不带时间信息。

### 【举例】

# 配置发往日志主机的日志信息的时间戳为 no-year-date 格式。

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

```
[Sysname] info-center timestamp loghost no-year-date
```

### 【相关命令】

- **info-center timestamp**

## 1.1.37 info-center trace-logfile quota

**info-center trace-logfile quota** 命令用来配置调试跟踪日志文件最大能占用的存储空间的大小。

**undo info-center trace-logfile quota** 命令用来恢复缺省情况。

### 【命令】

```
info-center trace-logfile quota size
```

```
undo info-center trace-logfile quota
```

### 【缺省情况】

调试跟踪日志文件最大能占用的存储空间的大小为 1MB。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**size**: 调试跟踪日志文件最大能占用的存储空间的大小，取值范围为 1~10，单位为 MB。

### 【举例】

# 配置调试跟踪日志文件最大能占用的存储空间的大小为 6MB。

```
<Sysname> system-view
[Sysname] info-center trace-logfile quota 6
```

## 1.1.38 logfile save

**logfile save** 命令用来手动将日志文件缓冲区中的内容全部保存到日志文件。

### 【命令】

**logfile save**

### 【视图】

任意视图

### 【缺省用户角色】

network-admin

### 【使用指导】

日志文件的保存路径可以通过 **info-center logfile directory** 命令配置。

日志文件保存成功后，日志文件缓冲区中的内容会被清空。

如果执行本命令时，日志文件缓冲区中没有日志，本命令会执行成功，但不会创建或更新已有日志文件。

### 【举例】

# 手动将日志文件缓冲区中的内容保存到日志文件。

```
<Sysname> logfile save
The contents in the log file buffer have been saved to the file flash:/logfile/logfile.log.
```

### 【相关命令】

- **info-center logfile enable**
- **info-center logfile directory**

## 1.1.39 reset logbuffer

**reset logbuffer** 命令用来清除日志缓冲区中的信息。

### 【命令】

**reset logbuffer**

### 【视图】

用户视图

### 【缺省用户角色】

network-admin

### 【举例】

# 清除日志缓冲区中的信息。

```
<Sysname> reset logbuffer
```

### 【相关命令】

- `display logbuffer`

#### 1.1.40 security-logfile save

`security-logfile save` 命令用来手动将安全日志文件缓冲区中的内容全部保存到安全日志文件。

### 【命令】

`security-logfile save`

### 【视图】

任意视图

### 【缺省用户角色】

`security-audit`

### 【使用指导】

安全日志文件保存成功后，安全日志文件缓冲区中的内容会被立即清空。

需要注意的是，只有具有安全日志管理员角色的本地用户才能使用本命令。安全日志管理员的配置请参见“安全命令参考”中的“AAA”。

如果执行本命令时，安全日志文件缓冲区中没有日志，本命令会执行成功，但不会创建或更新已有安全日志文件。

### 【举例】

# 手动将安全日志缓冲区中的内容保存到安全日志文件。

```
<Sysname> security-logfile save
```

```
The contents in the security log file buffer have been saved to the file  
flash:/seclog/seclog.log.
```

### 【相关命令】

- `info-center security-logfile switch-directory`
- `authorization-attribute`（安全命令参考/AAA）

#### 1.1.41 snmp-agent trap enable syslog

`snmp-agent trap enable syslog` 命令用来将系统日志封装成告警信息发送。

`undo snmp-agent trap enable syslog` 命令用来恢复缺省情况。

### 【命令】

`snmp-agent trap enable syslog`

`undo snmp-agent trap enable syslog`

### 【缺省情况】

系统日志不会封装成告警信息发送。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【使用指导】

配置该命令后，设备除了根据输出规则将日志输出到各方向外，还会将日志信息封装成告警信息，发送到 **SNMP** 模块和日志告警缓冲区。

- 发送到 **SNMP** 模块后，是否输出到目的主机，以 **Trap** 还是 **Inform** 形式发送，由 **SNMP** 模块的配置决定，关于告警信息以及 **SNMP** 配置的介绍请参见“网络管理和监控配置指导”中的“**SNMP**”。
- 发送到日志告警缓冲区后，用户可通过“日志告警缓冲区对应的 **MIB** 节点”来直接读取日志信息封装成的告警信息。用户还可通过 **info-center syslog trap buffersize** 命令来配置日志告警缓冲区的大小。

### 【举例】

```
# 将系统日志封装成告警信息发送。  
<Sysname> system-view  
[Sysname] snmp-agent trap enable syslog
```

### 【相关命令】

- **info-center syslog trap buffersize**

## 1.1.42 terminal debugging

**terminal debugging** 命令用来开启当前终端对调试信息的显示功能。

**undo terminal debugging** 命令用来关闭当前终端对调试信息的显示功能。

### 【命令】

```
terminal debugging  
undo terminal debugging
```

### 【缺省情况】

当前终端对调试信息的显示功能处于关闭状态。

### 【视图】

用户视图

### 【缺省用户角色】

network-admin

### 【使用指导】

如果需要在控制台显示调试信息，请先配置 **terminal debugging** 命令，再使能信息中心功能（信息中心功能缺省处于使能状态），最后使用 **debugging** 命令打开功能模块的调试信息开关。

如果需要在监视终端上显示调试信息，请先配置 **terminal monitor** 和 **terminal debugging** 命令，再使能信息中心功能（信息中心功能缺省处于使能状态），最后使用 **debugging** 命令打开功能模块的调试信息开关。

需要注意的是，本命令只对当前连接有效。当终端与设备重新建立连接后，本命令会恢复到缺省情况。

执行 **terminal logging level 7** 命令或 **terminal debugging** 命令，都可以开启当前终端对调试信息的显示功能，但两条命令有如下区别：

- 执行 **terminal logging level 7** 命令，当前终端允许输出级别为 0~7 的所有日志。
- 执行 **terminal debugging** 命令，当前终端仅输出 **terminal logging level** 命令配置的日志信息和级别为 7 的调试信息。

#### 【举例】

# 允许 debugging 日志信息输出到当前终端。

```
<Sysname> terminal debugging
```

```
The current terminal is enabled to display debugging information.
```

#### 【相关命令】

- **terminal logging level**
- **terminal monitor**

### 1.1.43 terminal logging level

**terminal logging level** 命令用来配置当前终端允许输出的日志信息的最低级别。

**undo terminal logging level** 命令用来恢复缺省情况。

#### 【命令】

```
terminal logging level severity
```

```
undo terminal logging level
```

#### 【缺省情况】

当前终端允许输出的日志信息的最低级别均为 6（Informational）。

#### 【视图】

用户视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**severity**: 当前终端允许输出的日志信息的最低级别，取值为 0~7 的整数或者 **alert**、**critical**、**debugging**、**emergency**、**error**、**informational**、**notification**、**warning**。

#### 【使用指导】

在配置了当前终端允许输出的日志信息的最低级别后，当系统输出信息时，所有信息等级高于或等于配置等级的信息都会被输出。例如，当配置的允许输出的日志信息的最低级别 6（informational）时，等级 0~6 的信息均会被输出。

本命令配置的显示属性只对当前连接有效，当终端与设备的连接超时、重新建立连接后，显示属性将恢复到缺省情况。

### 【举例】

# 配置当前终端监控的最低日志级别为 `debugging`。

```
<Sysname> terminal logging level 7
```

## 1.1.44 terminal monitor

**terminal monitor** 命令用来允许日志信息输出到当前终端。

**undo terminal monitor** 命令用来禁止日志信息输出到当前终端。

### 【命令】

```
terminal monitor
```

```
undo terminal monitor
```

### 【缺省情况】

允许日志信息输出到控制台，不允许日志信息输出到监视终端。

### 【视图】

用户视图

### 【缺省用户角色】

network-admin

### 【使用指导】

本命令只对当前连接有效。当终端与设备重新建立连接后，本命令会恢复到缺省情况。

### 【举例】

# 允许日志信息输出到当前终端。

```
<Sysname> terminal monitor
```

```
The current terminal is enabled to display logs.
```

# 目 录

1 Packet Capture .....	1-1
1.1 Packet Capture配置命令 .....	1-1
1.1.1 packet-capture interface.....	1-1
1.1.2 packet-capture read .....	1-3

# 1 Packet Capture

## 1.1 Packet Capture配置命令

### 1.1.1 packet-capture interface



说明

如需使用本命令，请先使用 **boot-loader** 命令安装 Packet Capture 特性软件包，有关安装步骤的详细介绍，请参见“基础配置指导”中的“软件升级”。

**packet-capture interface** 命令用来配置接口的入方向报文捕获。

#### 【命令】

捕获并保存报文到文件：

```
packet-capture interface interface-type interface-number [ capture-filter capt-expression | limit-captured-frames limit | limit-frame-size bytes | autostop filesize kilobytes | autostop duration seconds | autostop files numbers | capture-ring-buffer filesize kilobytes | capture-ring-buffer duration seconds | capture-ring-buffer files numbers ] * write filepath [ raw | { brief | verbose } ] *
```

捕获并显示报文内容：

```
packet-capture interface interface-type interface-number [ capture-filter capt-expression | display-filter disp-expression | limit-captured-frames limit | limit-frame-size bytes | autostop duration seconds ] * [ raw | { brief | verbose } ] *
```

#### 【视图】

用户视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**interface-type interface-number**：表示以太网接口的接口类型和接口编号，用来开启指定接口的报文捕获功能。

**capture-filter** *capt-expression*：表示捕获报文时过滤规则的表达式，为 1~256 个字符的字符串，区分大小写。如果不指定此参数，则捕获该接口的所有入方向的报文。

**display-filter** *disp-expression*：表示显示捕获报文时过滤规则的表达式，为 1~256 个字符的字符串，区分大小写。如果不指定此参数，则显示所有捕获到的报文。

**limit-captured-frames** *limit*：表示捕获报文的最大个数，取值范围为 0~2147483647，单位为个，缺省值为 10。若指定报文最大个数为 0，则表示没有限制。



**limit-frame-size bytes:** 表示捕获报文的最大长度，取值范围为 64~8000，单位为字节，缺省值为 8000。当捕获到的报文超过此长度，会对报文进行截断。

**autostop filesize kilobytes:** 表示报文文件（用来存储捕获报文的文件）的大小，取值范围为 1~65536，单位为千字节。如果没有指定本参数，表示对报文文件大小没有限制。

**autostop duration seconds:** 表示持续捕获报文的时长，取值范围为 1~2147483647，单位为秒。如果没有指定本参数，表示不对捕获报文的时长进行限制。

**autostop files numbers:** 表示允许写的报文文件的个数，取值范围为 2~64。如果没有指定本参数，表示不对允许写的报文文件的个数进行限制。

**capture-ring-buffer filesize kilobytes:** 表示切换存储报文文件大小，取值范围为 1~65536，单位为千字节。如果没有指定本参数，表示不以文件大小为限制切换报文文件。

**capture-ring-buffer duration seconds:** 表示切换存储报文文件时长，取值范围为 1~2147483647，单位为秒。如果没有本参数，表示不以时长为限制切换报文文件。

**capture-ring-buffer files numbers:** 表示报文文件的最大个数，取值范围为 2~64。如果没有指定本参数，表示报文文件的最大个数没有限制。

**write filepath:** 表示报文文件的名称，为 1~64 字符的字符串，后缀必须为“.pcap”，区分大小写。文件名命名规则的详细介绍，请参见“基础配置指导”中的“文件系统管理”。如果没有指定此参数，将不会保存捕获的报文。

**raw:** 将报文内容以十六进制格式显示。不指定该参数时，以字符串格式显示报文内容。

**verbose:** 显示捕获报文的详细信息。

**brief:** 显示捕获报文的简要信息。

【使用指导】

开启指定接口的报文捕获功能后，命令行输入界面会实时显示捕获报文的数量，此时，该用户界面下不允许输入命令行。如果用户希望停止捕获，直接输入 **Ctrl+C** 停止捕获报文。

当不指定 **raw**、**brief** 和 **verbose** 中的任何一个参数时，指定 **write** 参数，显示捕获的报文个数；当不指定 **raw**、**brief**、**verbose** 和 **write** 中的任何一个参数时，显示报文的简要信息。

当没有设置报文文件切换条件时，捕获的报文会保存在一个报文文件中，文件名称由 **write filepath** 参数指定。如果配置了报文文件切换条件，则捕获的报文会被保存到文件名为扩展文件名的文件中，扩展文件名由 **filepath**、文件生成序号和写入时间组成。当切换到新文件时，新生成的扩展报文文件序号按序递增。例如，指定的文件名称为 **a.pcap**，则第一个生成的报文文件的名称为 **a\_00001\_20140211034151.pcap**，当达到切换写文件条件时，则将报文写入新生成的 **a\_00002\_20140211034207.pcap** 文件中，依次类推。如果报文文件的最大个数已经达到，但是停止捕获条件还没有达到，设备会继续捕获报文，并用新捕获的报文覆盖生成时间最早的报文文件。

表1-1 packet-capture 命令参数描述表

操作	参数	说明
过滤表达式	<ul style="list-style-type: none"><li><b>capture-filter capt-expression:</b> 设备根据此规则对报文进行过滤并捕获匹配过滤规则的报文。</li><li><b>display-filter disp-expression:</b> 设备对已捕获的报文进行过滤，并将匹配的报文内容进行显示。</li></ul>	<p>捕获过滤不能对报文内容进行过滤，显示过滤可以对报文内容进行过滤</p> <p>捕获过滤规则详细描述请参见“网络管理和监控配置指导”中的“Packet Capture”。</p>

操作	参数	说明
停止捕获	<p>使用以下任意参数，即可实现设备自动停止捕获报文。当同时配置多个停止捕获参数时，先匹配成功的参数生效：</p> <ul style="list-style-type: none"> <li>• <b>autostop filesize kilobytes</b>: 当报文文件的大小达到配置值时，则自动停止捕获报文</li> <li>• <b>autostop duration seconds</b>: 当持续捕获报文的时长达到最大值时，则自动停止捕获报文</li> <li>• <b>autostop files numbers</b>: 当写过的文件的个数达到配置值时（包括被覆盖写过的文件），则自动停止捕获报文</li> <li>• <b>limit-captured-frames limit</b>: 当捕获报文的个数达到配置值时，则自动停止捕获报文</li> </ul>	<p>报文捕获功能还受文件系统空闲空间大小的限制。当文件系统空闲空间不足时，即便停止捕获参数设置的条件没有匹配，也将自动停止捕获报文</p>
切换存储	<p>使用以下任意参数，即可触发报文文件的切换。当同时配置多个切换参数时，先匹配成功的参数生效：</p> <ul style="list-style-type: none"> <li>• <b>capture-ring-buffer filesize kilobytes</b>: 当报文文件大小达到配置值时，切换到下一个文件来存储捕获报文</li> <li>• <b>capture-ring-buffer duration seconds</b>: 当捕获报文时长达到配置值时，切换到下一个文件来存储捕获报文</li> <li>• 当指定 <b>autostop files</b> 参数或者 <b>capture-ring-buffer files</b> 参数时，如果同时指定 <b>autostop filesize</b> 参数，则 <b>autostop filesize</b> 作为切换条件参数处理</li> </ul>	<p>当指定 <b>autostop files</b> 参数或者 <b>capture-ring-buffer files</b> 参数时，则需指定切换存储参数</p> <p>当同时指定 <b>autostop filesize</b> 和 <b>capture-ring-buffer filesize</b> 时，<b>autostop filesize</b> 作为停止条件参数将失效，<b>autostop filesize</b> 作为切换条件参数将生效。当 <b>autostop filesize</b> 和 <b>capture-ring-buffer filesize</b> 均作为切换条件参数，配置命令后输入的生效</p>

### 【举例】

```
# 配置接口 GigabitEthernet1/0/1 的入方向报文捕获。
<Sysname> packet-capture interface gigabitethernet 1/0/1
```

### 【相关命令】

- **packet-capture read**

## 1.1.2 packet-capture read



说明

如需使用本命令，请先使用 **boot-loader** 命令安装 Packet Capture 特性软件包，有关安装步骤的详细介绍，请参见“基础配置指导”中的“软件升级”。

**packet-capture read** 命令用来解析并显示保存的报文文件。

### 【命令】

```
packet-capture read filepath [ display-filter disp-expression ] [ raw |
{ brief | verbose } ] *
```

## 【视图】

用户视图

## 【缺省用户角色】

network-admin

## 【参数】

**filepath:** 指定读取的文件的完整路径, 为 1~64 个字符的字符串, 后缀为“.pcap”或“.pcapng”, 区分大小写。文件名命名规则的详细介绍, 请参见“基础配置指导”中的“文件系统管理”。

**display-filter disp-expression:** 指定用来显示报文的过滤规则, 为 1~256 个字符的字符串。*disp-expression* 为显示报文的过滤规则。设备报文文件内容匹配参数指定的显示过滤规则, 并将匹配的报文内容进行显示。显示过滤语法规则参见 **Packet Capture** 配置手册。如果不指定此参数, 则显示报文文件中的所有报文信息。

**raw:** 将报文文件内容用十六进制格式显示。不指定该参数时, 以字符串格式显示报文内容。

**brief:** 显示报文文件的简要信息。

**verbose:** 显示报文文件的详细信息。

## 【使用指导】

配置解析并显示报文文件内容后, **Packet Capture** 终端显示从指定文件中读取解析的报文信息, 如果用户希望退出此过程, 可以直接输入 **Ctrl+C** 退出解析过程。

设备保存报文文件时使用 **pcap** 格式, 但解析的报文文件可以为 **pcap** 或 **pcapng** 格式。

未指定 **raw**、**brief**、**verbose** 参数时, 则显示简要信息。

## 【举例】

# 解析 flash:/test 目录下的报文文件 aaaa.pcap。

```
<Sysname> packet-capture read flash:/test/aaaa.pcap
```

## 【相关命令】

- **packet-capture interface**

# 目 录

1 云平台连接.....	1-1
1.1 云平台连接配置命令.....	1-1
1.1.1 cloud-management keepalive.....	1-1
1.1.2 cloud-management server domain.....	1-1
1.1.3 display cloud-management state.....	1-2
1.1.4 reset cloud-management tunnel.....	1-4

# 1 云平台连接

## 1.1 云平台连接配置命令

### 1.1.1 cloud-management keepalive

**cloud-management keepalive** 命令用来配置向云平台服务器发送 Keepalive 报文的时间间隔。

**undo cloud-management keepalive** 命令用来恢复缺省情况。

#### 【命令】

```
cloud-management keepalive interval  
undo cloud-management keepalive
```

#### 【缺省情况】

设备向云平台服务器发送 Keepalive 报文的时间间隔为 180 秒。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**interval**: 发送 Keepalive 报文的时间间隔，取值范围为 180~600，单位为秒。

#### 【使用指导】

设备与云平台服务器建立连接后，会周期性地向服务器发送 Keepalive 报文进行保活。如果设备在连续 3 个 Keepalive 报文的发送周期内没有收到云平台服务器的响应，设备会重新向云平台服务器发送认证请求，与其重新建立连接。

#### 【举例】

# 配置向云平台服务器发送 Keepalive 报文的时间间隔为 360 秒。

```
<Sysname> system-view  
[Sysname] cloud-management keepalive 360
```

### 1.1.2 cloud-management server domain

**cloud-management server domain** 命令用来指定云平台服务器域名。

**undo cloud-management server domain** 命令用来恢复缺省情况。

#### 【命令】

```
cloud-management server domain domain-name  
undo cloud-management server domain
```

#### 【缺省情况】

未指定云平台服务器域名。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*domain-name*: 云平台服务器的域名，为 1~253 个字符的字符串，区分大小写。

### 【使用指导】

配置云平台服务器域名时，请确认网络中已经配置了能够解析该域名的 DNS 服务器。

只能指定一个云平台服务器域名，多次执行本命令，最后一次执行的命令生效。

### 【举例】

# 配置云平台服务器域名为 abc.com。

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

```
[Sysname] cloud-management server domain abc.com
```

## 1.1.3 display cloud-management state

**display cloud-management state** 命令用来显示云平台连接的状态。

### 【命令】

**display cloud-management state**

### 【视图】

任意视图

### 【缺省用户角色】

network-admin

network-operator

### 【举例】

# 显示云平台连接的状态。

```
<Sysname> display cloud-management state
```

```
Cloud connection state    : Established
```

```
Device state              : Request_success
```

```
Cloud server address      : 10.1.1.1
```

```
Cloud server domain name  : abc.com
```

```
Cloud server port         : 443
```

```
Connected at              : Wed Jan 27 14:18:40 2016
```

```
Duration                  : 00d 00h 02m 01s
```

```
Process state             : Message received
```

```
Failure reason            : N/A
```

表1-1 display cloud-management state 命令显示信息描述表

字段	描述
Cloud connection state	<p>云平台连接状态，包括以下取值：</p> <ul style="list-style-type: none"> <li>• Unconnected: 未建立连接</li> <li>• Request: 已发送连接请求</li> <li>• Established: 已建立连接</li> </ul>
Device state	<p>设备状态，包括以下取值：</p> <ul style="list-style-type: none"> <li>• Idle: 空闲</li> <li>• Connecting: 与云平台服务器连接中</li> <li>• Request_CAS_url: 已发送请求 CAS（Central Authentication Service，中央认证服务）服务器 URL 的报文</li> <li>• Request_CAS_url_success: 请求 CAS 服务器的 URL 成功</li> <li>• Request_CAS_TGT: 已发送请求 CAS 认证所需 TGT（Ticket Granting Ticket，票据授权票据）的报文</li> <li>• Request_CAS_TGT_success: 请求 CAS 认证所需 TGT 成功</li> <li>• Request_CAS_ST: 已发送请求 CAS 认证所需 ST（Service Ticket，服务票据）的报文</li> <li>• Request_CAS_ST_success: 请求 CAS 认证所需 ST 成功</li> <li>• Request_cloud_auth: 已向云平台服务器发送云平台认证请求</li> <li>• Request_cloud_auth_success: 已通过云平台认证</li> <li>• Register: 向云平台服务器发起注册</li> <li>• Register_success: 在云平台服务器上注册成功</li> <li>• Request: 已发送握手请求</li> <li>• Request_success: 握手成功</li> </ul>
Cloud server address	云平台服务器地址
Cloud server domain name	云平台服务器域名
Cloud server port	与云平台服务器建立连接所使用的TCP端口号
Connected at	云平台连接的建立时间
Duration	云平台连接的持续时长，格式为xx天xx时xx分xx秒
Process state	<p>云连接流程状态，包括以下取值：</p> <ul style="list-style-type: none"> <li>• DNS not parsed: 未解析 DNS</li> <li>• DNS parsed: 解析 DNS 完成</li> <li>• Message not sent: 未发送消息</li> <li>• Message sent: 已发送消息</li> <li>• Message not received: 未接收消息</li> <li>• Message received: 已接收消息</li> </ul>

字段	描述
Failure reason	<p>云连接失败原因，包括以下取值：</p> <ul style="list-style-type: none"> <li>• DNS parse failed: DNS 解析失败</li> <li>• Socket connection failed: Socket 连接失败</li> <li>• SSL creation failed: SSL 模式创建失败</li> <li>• Sending CAS url request failed: 发送请求 CAS（Central Authentication Service，中央认证服务）服务器 URL 的报文失败</li> <li>• Sending CAS TGT failed: 发送请求 CAS 认证所需 TGT（Ticket Granting Ticket，票据授权票据）的报文失败</li> <li>• Sending CAS ST failed: 发送请求 CAS 认证所需 ST（Service Ticket，服务票据）的报文失败</li> <li>• Sending cloud auth request failed: 向云平台服务器发送云平台认证请求失败</li> <li>• Sending registration request failed: 向云平台服务器发送注册请求失败</li> <li>• Processing CAS url response failed: 处理 CAS 服务器 URL 的回应报文失败</li> <li>• Processing CAS TGT response failed: 处理 CAS 认证所需 TGT 的回应报文失败</li> <li>• Processing CAS ST response failed: 处理 CAS 认证所需 ST 的回应报文失败</li> <li>• Processing cloud auth response failed: 处理云平台认证回应报文失败</li> <li>• Processing registration response failed: 处理云平台服务器注册回应报文失败</li> <li>• Sending handshake request failed: 发送握手请求报文失败</li> <li>• Processing handshake failed: 处理握手报文失败</li> <li>• Sending websocket request failed: 发送 websocket 请求失败</li> <li>• Processing websocket packet failed: 处理 websocket 报文失败</li> </ul>

#### 1.1.4 reset cloud-management tunnel

`reset cloud-management tunnel` 命令用来重新建立设备与云平台的连接。

##### 【命令】

```
reset cloud-management tunnel
```

##### 【视图】

用户视图

##### 【缺省用户角色】

network-admin

##### 【举例】

# 重建设备与云平台的连接。

```
<Sysname> reset cloud-management tunnel
```



# 目 录

1 SmartMC .....	1-1
1.1 SmartMC配置命令 .....	1-1
1.1.1 boot-loader .....	1-1
1.1.2 create batch-file .....	1-2
1.1.3 display smartmc backup configuration status .....	1-3
1.1.4 display smartmc batch-file status .....	1-4
1.1.5 display smartmc configuration .....	1-5
1.1.6 display smartmc device-link .....	1-6
1.1.7 display smartmc group .....	1-7
1.1.8 display smartmc replace status .....	1-8
1.1.9 display smartmc resource-monitor .....	1-9
1.1.10 display smartmc resource-monitor configuration .....	1-11
1.1.11 display smartmc tc .....	1-12
1.1.12 display smartmc tc log buffer .....	1-14
1.1.13 display smartmc tc log restart .....	1-15
1.1.14 display smartmc upgrade status .....	1-16
1.1.15 display smartmc vlan .....	1-17
1.1.16 match .....	1-17
1.1.17 smartmc auto-link-aggregation enable .....	1-18
1.1.18 smartmc auto-replace enable .....	1-19
1.1.19 smartmc backup configuration .....	1-20
1.1.20 smartmc backup configuration max-number .....	1-20
1.1.21 smartmc backup configuration interval .....	1-21
1.1.22 smartmc batch-file apply .....	1-22
1.1.23 smartmc batch-file deploy .....	1-23
1.1.24 smartmc enable .....	1-24
1.1.25 smartmc ftp-server .....	1-25
1.1.26 smartmc group .....	1-25
1.1.27 smartmc outbound .....	1-26
1.1.28 smartmc resource-monitor .....	1-27
1.1.29 smartmc resource-monitor interval .....	1-28
1.1.30 smartmc resource-monitor max-age .....	1-28
1.1.31 smartmc replace .....	1-29

1.1.32 smartmc tc boot-loader.....	1-30
1.1.33 smartmc tc device-type .....	1-30
1.1.34 smartmc tc password .....	1-31
1.1.35 smartmc tc startup-configuration.....	1-31
1.1.36 smartmc topology-refresh .....	1-32
1.1.37 smartmc topology-refresh interval .....	1-32
1.1.38 smartmc topology-save .....	1-33
1.1.39 smartmc upgrade boot-loader .....	1-34
1.1.40 smartmc upgrade startup-configuration .....	1-35
1.1.41 smartmc vlan .....	1-36
1.1.42 startup-configuration .....	1-37

# 1 SmartMC



## 说明

- 仅 Release 6127 及以上版本支持 SmartMC 特性。
- S5110V2-SI、S5000V3-EI 和 S5000E-X 系列交换机不支持 SmartMC 特性。
- S5130S-SI[LI]、S5120V2-SI[LI]和 S3100V3-SI 系列交换机仅支持作为 SmartMC 网络中的成员设备。

## 1.1 SmartMC配置命令

### 1.1.1 boot-loader

**boot-loader** 命令用来配置 SmartMC 组升级使用的启动软件。

**undo boot-loader** 命令用来恢复缺省情况。

#### 【命令】

```
boot-loader file { ipe-filename | boot boot-filename system
system-filename }
undo boot-loader
```

#### 【缺省情况】

未配置 SmartMC 组升级使用的启动软件。

#### 【视图】

SmartMC 组视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**ipe-filename**: 指定 SmartMC 组升级使用的 IPE 文件，为 5~45 个字符的字符串，不区分大小写，以.ipe 为后缀名。

**boot boot-filename**: 指定 SmartMC 组升级使用的 Boot 文件，为 5~45 个字符的字符串，不区分大小写，以.bin 为后缀。

**system system-filename**: 指定 SmartMC 组升级使用的 System 文件，为 5~45 个字符的字符串，不区分大小写，以.bin 为后缀。

#### 【使用指导】

多次执行该命令，最后一次执行的命令生效。

#### 【举例】

# 配置 SmartMC 组 testgroup 升级使用的 IPE 文件为 device.ipe。

```
<Sysname> system-view
[Sysname] smartmc group testgroup
[Sysname-smartmc-group-testgroup] boot-loader file device.ipe
```

#### 【相关命令】

- **smartmc group**
- **smartmc upgrade boot-loader**

### 1.1.2 create batch-file

**create batch-file** 命令用来创建命令行批处理文件。

#### 【命令】

```
create batch-file batch-file-name
```

#### 【缺省情况】

不存在命令行批处理文件。

#### 【视图】

用户视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*batch-file-name*: 命令行批处理文件名称，为 1~255 个字符的字符串，不区分大小写，如果用户不输入后缀名，则文件名后缀缺省为.cmdset。

#### 【使用指导】

执行该命令后，设备进入批量编辑命令行模式。在该模式下编辑命令时，每条命令占用一行，编辑完成后，输入“%”和回车结束编辑，并退回用户视图。

设备不检查命令行的正确性，因此，在编辑命令行时，请用户保证命令行的正确性。

#### 【举例】

# 编辑名称为 test.cmdset 的命令行批处理文件，配置设备名称为 Sysname，并开启 Telnet 功能。

```
<Sysname> create batch-file test.cmdset
Begin to edit batch commands, and quit with the character '%'.
system-view
sysname Sysname
telnet server enable%
<Sysname>
```

#### 【相关命令】

- **display smartmc batch-file status**
- **smartmc batch-file deploy**

### 1.1.3 display smartmc backup configuration status

**display smartmc backup configuration status** 命令用来显示成员设备备份配置文件的状态。

#### 【命令】

**display smartmc backup configuration status**

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

#### 【使用指导】

执行本命令时：

- 如果有成员设备正在进行备份操作，则显示正在进行备份操作的成员设备的备份状态。
- 如果没有成员设备进行备份操作，则显示最近一次执行备份操作的成员设备的备份状态。

#### 【举例】

# 显示正在进行备份操作的成员设备的备份状态。

```
<Sysname> display smartmc backup configuration status
```

ID	IpAddress	MacAddress	Status	Time
1	192.168.56.30	08d2-38ff-0300	Finished	2017-04-05 11:30:35
2	192.168.56.40	62d2-c21c-0400	Finished	2017-04-05 11:30:40

表1-1 display smartmc backup configuration status 命令显示信息描述表

字段	描述
ID	成员设备的编号
MacAddress	成员设备的MAC地址
IpAddress	成员设备的IP地址
Status	备份状态： <ul style="list-style-type: none"><li>• Waiting: 表示设备处于等待备份状态</li><li>• Processing: 表示设备处于正在备份状态</li><li>• Finished: 表示设备备份完成</li><li>• Timeout: 表示设备备份超时</li><li>• Failed: 表示设备备份失败</li></ul>
Time	完成备份的时间，如果成员设备没有完成备份，则显示为“-”

#### 【相关命令】

- smartmc backup configuration
- smartmc backup configuration interval
- smartmc backup configuration max-number

### 1.1.4 display smartmc batch-file status

**display smartmc batch-file status** 命令用来显示执行命令行批处理文件的结果。

#### 【命令】

**display smartmc batch-file status** [ **ap** | **last number** | **phone** ]

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**ap**: 显示最近一次连接 AP 的端口执行命令行批处理文件的结果。

**last number**: 显示最近第几次管理设备通过 **smartmc batch-file deploy** 命令向成员设备下发批处理命令的执行结果，取值范围为 1~5。

**phone**: 显示最近一次连接 IP 电话的端口执行命令行批处理文件的结果。

#### 【使用指导】

如果不指定任何参数，则显示最近一次管理设备通过 **smartmc batch-file deploy** 命令向成员设备或 SmartMC 组批量下发命令行的执行结果。

#### 【举例】

# 显示最近一次批量命令行的执行结果（命令行批处理文件中包含 **display smartmc configuration** 命令）。

```
<Sysname> display smartmc batch-file status last 1
```

```
TC ID 1
```

```
Device MAC : 8a73-60c3-0200
```

```
Start Time : 2018-12-24 14:55:39
```

```
End Time : 2018-12-24 14:55:43
```

```
Result :
```

```
<Sysname>display smartmc configuration
```

```
Device role : TC
```

```
TM IP : 192.168.22.103
```

```
TM MAC : 8a73-4faa-0100
```

```
TM sysname : Sysname
```

```
<Sysname>
```

```
TC ID 2
```

```
Device MAC : 8a73-6b31-0300
```

```
Start Time : 2018-12-24 14:55:43
```

```
End Time : 2018-12-24 14:55:48
```

```
Result :
```

```
<Sysname>display smartmc configuration
```

```
Device role : TC
```

```

TM IP                : 192.168.22.103
TM MAC               : 8a73-4faa-0100
TM sysname           : Sysname
<Sysname>

```

表1-2 display smartmc batch-file status 命令显示信息描述表

字段	描述
TC ID	成员设备的编号
Device MAC	成员设备的MAC地址
Start Time	执行命令行批处理文件的开始时间
End Time	执行命令行批处理文件的结束时间
Result	执行命令行批处理文件的详细结果

### 【相关命令】

- **create batch-file**
- **smartmc batch-file apply**
- **smartmc batch-file deploy**

## 1.1.5 display smartmc configuration

**display smartmc configuration** 命令用来显示 SmartMC 功能的配置信息。

### 【命令】

```
display smartmc configuration
```

### 【视图】

任意视图

### 【缺省用户角色】

network-admin

### 【举例】

# 在管理设备上显示 SmartMC 功能的配置信息。

```

<Sysname> display smartmc configuration
Device role                : TM
FTP server IP              : 192.168.22.103
FTP server username        : admin
Topology-refresh interval  : 60(s)
Backup startup-configuration interval : N/A
Sync backup number         : 5
Device status              : Lack
Some configurations are absent on the TM, such as Telnet or LLDP configuration.

```

# 在成员设备上显示管理设备的信息。

```
<Sysname> display smartmc configuration
```

```

Device role      : TC
TM IP           : 192.168.22.103
TM MAC          : 8288-468d-0100
TM sysname      : Sysname

```

表1-3 display smartmc configuration 命令显示信息描述表

字段	描述
Device role	当前设备的角色
FTP server IP	FTP服务器的IP地址，若没有配置，则显示为N/A
FTP server username	FTP服务器的用户名，若没有配置，则显示为N/A
Topology-refresh interval	拓扑刷新间隔
Backup startup-configuration interval	自动备份配置文件的时间间隔，单位为小时，若没有配置，则显示为N/A
Sync backup number	同时进行备份操作的成员设备的数量
Device status	管理设备状态 <ul style="list-style-type: none"> <li>• Normal: 正常</li> <li>• Lack: 配置缺失（NETCONF, Telnet, Local User, LLDP）</li> </ul>
TM IP	管理设备的IP地址，如果没有获取到管理设备的IP地址，则显示为N/A
TM MAC	管理设备的MAC地址，如果没有获取到管理设备的MAC地址，则显示为N/A
TM sysname	管理设备的名称，如果没有获取到管理设备的名称，则显示为N/A
Some configurations are absent on the TM, such as XXX.	管理设备上配置缺失。如果Device status字段取值为“Lack”，则显示本字段，提示管理设备上缺少相关配置。配置缺失会影响SmartMC相关功能的正常运行，请根据提示完善相关配置

### 1.1.6 display smartmc device-link

**display smartmc device-link** 命令用来显示 SmartMC 网络中设备间的连接信息。

#### 【命令】

```
display smartmc device-link
```

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

#### 【举例】

# 显示 SmartMC 网络的拓扑信息。

```

<Sysname> display smartmc device-link
(TM IP)[192.168.56.20]

```

ID	Hop	LocalPort	LocalIP	PeerPort	PeerIP
0	0	GigabitEthernet1/0/2	192.168.56.20	GigabitEthernet1/0/1	192.168.56.30



1	1	GigabitEthernet1/0/1	192.168.56.30	GigabitEthernet1/0/2	192.168.56.20
1	2	GigabitEthernet1/0/2	192.168.56.30	GigabitEthernet1/0/1	192.168.56.40
2	3	GigabitEthernet1/0/1	192.168.56.40	GigabitEthernet1/0/2	192.168.56.30

表1-4 display smartmc device-link 命令显示信息描述表

字段	描述
TM IP	管理设备的IP地址
ID	管理设备和成员设备的ID
Hop	管理设备和成员设备之间的跳数
LocalPort	本地端口
LocalIP	本地设备的IP地址
PeerPort	对端端口
PeerIP	对端设备的IP地址

#### 【相关命令】

- **smartmc topology-refresh**
- **smartmc topology-refresh interval**

### 1.1.7 display smartmc group

**display smartmc group** 命令用来显示 SmartMC 组的信息。

#### 【命令】

**display smartmc group** [ *group-name* ] [ **verbose** ]

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**group-name**: SmartMC 组的名称，为 1~31 个字符的字符串，区分大小写。不指定该参数时，显示所有 SmartMC 组的信息。

**verbose**: 显示 SmartMC 组的详细信息。不指定该参数时，显示 SmartMC 组的简要信息。

#### 【举例】

# 显示 SmartMC 组的详细信息。

```
<Sysname> display smartmc group verbose
Group name           : test
TC count             : 3
Boot-loader file     :
Startup-configuration file :
Rule:
```

```

Match Device-type S5130S-LI
TCID DeviceType Sysname IpAddress MacAddress Status Version
1 S5130S-LI S1 192.168.56.103 0e74-e2fb-0400 Normal COMWAREV700R001
2 S5130S-LI S2 192.168.56.102 0e74-ea13-0500 Normal COMWAREV700R001
3 S5130S-LI S3 192.168.56.104 0e74-db54-0300 Normal COMWAREV700R001

```

表1-5 display smartmc group verbose 命令显示信息描述表

字段	描述
GroupName	SmartMC组的名称
TC count	SmartMC组内成员设备的数量
Boot-loader file	SmartMC组升级时使用的软件升级文件名称，取值为空表示未配置
Startup-configuration file	SmartMC组升级时使用的配置文件，取值为空表示未配置
Rule	SmartMC组的匹配规则
Match	匹配类型以及匹配类型值，匹配类型取值包括： <ul style="list-style-type: none"> <li>• <b>Device-type</b>: 表示按照设备类型匹配成员设备</li> <li>• <b>IP-address</b>: 表示按照 IP 地址匹配成员设备</li> <li>• <b>MAC-address</b>: 表示按照 MAC 地址匹配成员设备</li> </ul>
TCID	成员设备的编号
DeviceType	成员设备的设备类型
Sysname	设备名称
IpAddress	成员设备的IP地址
MacAddress	成员设备的MAC地址
Version	成员设备的软件版本号
Status	成员设备的运行状态： <ul style="list-style-type: none"> <li>• <b>Offline</b>: 表示成员设备处于离线状态</li> <li>• <b>Normal</b>: 表示成员设备正常上线</li> </ul>

#### 【相关命令】

- **match**
- **smartmc group**

### 1.1.8 display smartmc replace status

**display smartmc replace status** 命令用来显示新成员设备替换故障成员设备的替换状态。

#### 【命令】

**display smartmc replace status**

#### 【视图】

任意视图

【缺省用户角色】

network-admin

【举例】

# 显示新成员设备替换故障成员设备的状态。

```
<Sysname> display smartmc replace status
Faulty ID      : 2
Faulty MAC     : 94e2-cdcb-0600
Replacement ID : 3
Replacement MAC: 2443-5f8c-0200
Mode           : Manual
Status         : Successful
Start time     : 2017-03-21 15:01:31
End time       : 2017-03-21 15:01:40
```

表1-6 display smartmc replace status 命令显示信息描述表

字段	描述
Faulty ID	故障成员设备编号
Faulty MAC	故障成员设备的MAC地址
Replacement ID	新成员设备编号
Replacement MAC	新成员设备的MAC地址
Mode	故障替换方式： <ul style="list-style-type: none"><li>Manual: 表示手动替换</li><li>Auto: 表示自动替换</li></ul>
Status	替换状态： <ul style="list-style-type: none"><li>Successful: 表示替换成功</li><li>Failed: 表示替换失败</li><li>Replacing: 表示新成员设备正在替换故障成员设备</li><li>Timeout: 表示替换超时</li></ul>
Start time	替换开始时间
End time	替换结束时间

【相关命令】

- smartmc auto-replace enable
- smartmc replace

1.1.9 display smartmc resource-monitor

display smartmc resource-monitor 命令用来显示设备的资源监控信息。

### 【命令】

```
display smartmc resource-monitor [ cpu | memory | temperature ] * [ tc tc-id  
| tm ]
```

### 【视图】

任意视图

### 【缺省用户角色】

network-admin

### 【参数】

**cpu:** 显示 CPU 的使用率。

**memory:** 显示内存的使用率。

**temperature:** 显示成员设备的温度信息。

**tc tc-id:** 显示成员设备的资源监控信息，tc-id 为成员设备的编号，取值范围为 1~255。

**tm:** 显示管理设备的资源监控信息。

### 【使用指导】

如果不指定资源类型，则显示所有类型的资源监控信息。

如果不指定设备的角色，则显示管理设备和所有成员设备的资源监控信息。

### 【举例】

# 显示成员设备 1 的所有资源监控信息。

```
<Sysname> display smartmc resource-monitor tc 1  
TC 1  
Collection time : 2017-07-25 18:02:30  
Slot 1 :  
CPU 0 CPU usage: 1%  
Memory usage   : 587076/903332  
Temperature    : 30
```

表1-7 display smartmc resource-monitor 命令显示信息描述表

字段	描述
TC ID	成员设备的编号
Collection time	监控信息的采集时间
Slot ID	Slot的编号
CPU x CPU usage	CPU编号和CPU使用率
Memory usage	内存使用率
Temperature	温度信息

### 【相关命令】

- smartmc resource-monitor

### 1.1.10 display smartmc resource-monitor configuration

**display smartmc resource-monitor configuration** 命令用来显示资源监控功能的配置信息。

#### 【命令】

**display smartmc resource-monitor configuration**

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

#### 【举例】

# 显示资源监控功能的配置信息。

```
<Sysname> display smartmc resource-monitor configuration
```

ID	MacAddress	CPU	Memory	Temperature
1	1111-2222-3333	Y	N	N
2	1111-2222-3334	Y	N	N
3	1111-2222-3335	Y	N	N

表1-8 display smartmc resource-monitor configuration 命令显示信息描述表

字段	描述
ID	设备的编号
MacAddress	设备的MAC地址
CPU	CPU监控的开启状态： <ul style="list-style-type: none"><li>Y：该监控资源处于开启状态</li><li>N：该监控资源处于关闭状态</li><li>-：设备不支持该资源监控</li></ul>
Memory	Memory监控的开启状态： <ul style="list-style-type: none"><li>Y：该监控资源处于开启状态</li><li>N：该监控资源处于关闭状态</li><li>-：设备不支持该资源监控</li></ul>
Temperature	温度信息监控的开启状态： <ul style="list-style-type: none"><li>Y：该监控资源处于开启状态</li><li>N：该监控资源处于关闭状态</li><li>-：设备不支持该资源监控</li></ul>

#### 【相关命令】

- **smartmc resource-monitor**

### 1.1.11 display smartmc tc

**display smartmc tc** 命令用来显示成员设备的信息。

#### 【命令】

**display smartmc tc** [ *tc-id* ] [ **verbose** ]

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**tc-id**: 成员设备的编号，取值范围为 1~255。如果不指定该参数，则显示所有成员设备的信息。

**verbose**: 显示成员设备的详细信息。不指定该参数时，则显示成员设备的简要信息。

#### 【举例】

# 显示所有成员设备的简要信息。

```
<Sysname> display smartmc tc
TCID DeviceType Sysname  IPAddress      MacAddress      Status  Version
1     S5130S-LI  S1              192.168.22.104  201c-e7c3-0300 Normal  COMWAREV700R001
```

表1-9 display smartmc tc 命令显示信息描述表

字段	描述
TCID	成员设备的编号
DeviceType	成员设备的设备类型
Sysname	成员设备的名称
IpAddress	成员设备的IP地址
MacAddress	成员设备的MAC地址
Status	成员设备的运行状态：取值包括： <ul style="list-style-type: none"><li>• <b>Normal</b>：表示设备运行正常</li><li>• <b>Offline</b>：表示设备不在线</li></ul>
Version	成员设备的软件版本号

# 显示所有成员设备的详细信息

```
<Sysname> display smartmc tc verbose
TC ID                : 1
Adding method        : Manual
Sysname              : S1
Model                : S5130S-28S-PWR-LI
Device type          : S5130S-LI
SYSOID               : 1.3.6.1.4.1.25506
MAC address          : 0e74-e2fb-0400
IP address           : 192.168.56.103
```

```

Boot image          :
Boot image version  :
System image        :
System image version :
Current-configuration file :
Uptime              : 2 days, 3 hours, 4 minutes
System CPU usage    : 0%
System memory usage : 0%
Status              : Offline
Boot-loader file    :
Startup-configuration file :

```

表1-10 display smartmc tc verbose 命令显示信息描述表

字段	描述
TC ID	成员设备的编号
Adding method	成员设备加入到SmartMC网络的方式： <ul style="list-style-type: none"> <li>Manual: 表示手动加入 SmartMC 网络</li> <li>Auto: 表示自动加入 SmartMC 网络</li> </ul>
Sysname	成员设备的名称
Model	成员设备的设备型号
Device type	成员设备的设备类型
SYSOID	成员设备的SYSOID
MAC address	成员设备的MAC地址
IP address	成员设备的IP地址
Boot image	成员设备的运行boot文件
Boot image version	成员设备的运行boot文件版本
System image	成员设备的运行system文件
System image version	成员设备的运行system文件版本
Current-configuration file	成员设备当前使用的配置文件
Uptime	成员设备的上电运行时间
System CPU usage	成员设备当前的CPU使用率
System memory usage	成员设备当前的内存使用率
Status	成员设备的运行状态：取值包括： <ul style="list-style-type: none"> <li>Normal: 表示设备运行正常</li> <li>Offline: 表示设备不在线</li> </ul>
Boot-loader file	成员设备升级使用的软件升级文件
Startup-configuration file	成员设备升级使用的配置文件

## 1.1.12 display smartmc tc log buffer

**display smartmc tc log buffer** 命令用来显示成员设备的日志缓冲区中的日志信息。

### 【命令】

```
display smartmc tc tc-id log buffer [ module module-name [ mnemonic mnemonic-value ] ]
```

### 【视图】

任意视图

### 【缺省用户角色】

network-admin

### 【参数】

**tc-id**: 成员设备的编号，取值范围为 1~255。

**module** *module-name*: 日志信息的应用模块的名称，为 1~8 个字符的字符串，不区分大小写。可以执行 **info-center source** 显示模块名称。有关 **info-center source** 命令的详细介绍，请参见“网络管理和监控命令参考”中的“信息中心”。

**mnemonic** *mnemonic-value*: 日志信息的助记符，为 1~32 个字符的字符串，不区分大小写。

### 【举例】

# 显示 ID 为 1，助记符为 SHELL\_CMD 的成员设备的日志信息

```
<Sysname> display smartmc tc 1 log buffer module SHELL mnemonic SHELL_CMD
```

```
Time      : 2017-07-15 13:51:46
Level     : Informational
Module    : SHELL
Mnemonic  : SHELL_CMD
Content   : -Line=con0-IPAddr=**-User=**; Command is qu
```

```
Time      : 2017-07-15 13:51:39
Level     : Informational
Module    : SHELL
Mnemonic  : SHELL_CMD
Content   : -Line=con0-IPAddr=**-User=**; Command is local-user admin
```

表1-11 display smartmc tc log buffer 命令显示信息描述表

字段	描述
Time	日志生成时间
Level	日志等级
Module	日志模块名称
Mnemonic	日志助记符
Content	日志描述



### 1.1.13 display smartmc tc log restart

**display smartmc tc log restart** 命令显示成员设备重启的日志信息。

#### 【命令】

**display smartmc tc *tc-id* log restart**

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*tc-id*: 成员设备的编号，取值范围为 1~255。

#### 【使用指导】

成员设备除了将各业务模块生成的日志保存到自己的日志缓冲区以外，还会将系统重启的日志发送给管理设备，管理设备会为每个成员设备创建重启日志缓冲区，分别存储各成员设备重启的日志信息。

也可以通过 **display smartmc tc *tc-id* log module SYSLOG mnemonic SYSLOG\_RESTART** 命令查看成员设备重启的日志信息。

管理设备仅为每个成员设备保存 10 条重启日志，如果超过 10 条，则最新日志覆盖最旧日志信息。

#### 【举例】

# 显示 ID 为 1 的成员设备重启的日志信息。

```
<Sysname> display smartmc tc 1 log restart
Time      : 2017-07-15 13:51:46
Level     : Informational
Module    : SYSLOG
Mnemonic  : SYSLOG_RESTART
Content   : System restarted -- H3C Comware Software.
```

表1-12 display smartmc tc log restart 命令显示信息描述表

字段	描述
Time	日志生成时间
Level	日志等级
Module	日志模块名称
Mnemonic	日志助记符
Content	日志描述

#### 【相关命令】

- **display smartmc tc log buffer**

### 1.1.14 display smartmc upgrade status

**display smartmc upgrade status** 命令用来显示成员设备的升级状态。

#### 【命令】

**display smartmc upgrade status**

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

#### 【举例】

# 显示成员设备的升级状态。

```
<Sysname> display smartmc upgrade status
```

ID	IpAddress	MacAddress	Status	UpdateTime	UpdateFile
1	192.168.56.1	82dd-a434-0200	Finished	Immediately	bootloader.ipe
2	192.168.56.103	5caf-2e5f-0100	Finished	Immediately	bootloader.ipe

表1-13 display smartmc upgrade status 命令显示信息描述表

字段	描述
ID	成员设备的编号
MacAddress	成员设备的MAC地址
IpAddress	成员设备的IP地址
Status	升级状态： <ul style="list-style-type: none"><li>• <b>Waiting</b>: 表示成员设备正在等待下载升级文件</li><li>• <b>Download-failed</b>: 表示成员设备下载升级文件失败</li><li>• <b>Download-finished</b>: 表示成员设备下载升级文件完成</li><li>• <b>Downloading</b>: 表示成员设备正在下载升级文件</li><li>• <b>Updating</b>: 表示成员设备升级中</li><li>• <b>Finished</b>: 表示成员设备升级完成</li><li>• <b>Failed</b>: 表示成员设备升级失败</li><li>• <b>Unknown</b>: 表示成员设备升级状态未知</li></ul>
UpdateFile	升级文件的名称
UpdateTime	升级时间 <ul style="list-style-type: none"><li>• <b>Immediately</b>: 表示成员设备立即升级</li><li>• <b>Delay(m)</b>: 表示延时升级，及延时时间</li><li>• <b>Time(HH:MM)</b>: 表示定时升级，及定时升级的时间</li></ul>

#### 【相关命令】

- **smartmc upgrade group**
- **smartmc upgrade tc**

1.1.15 display smartmc vlan

display smartmc vlan 命令用来显示成员设备创建 VLAN 的结果。

【命令】

display smartmc vlan

【视图】

任意视图

【缺省用户角色】

network-admin

【举例】

# 显示成员设备创建 VLAN 的结果。

```
<Sysname> display smartmc vlan
ID      IpAddress      MacAddress      Vlan      Status
1       192.168.22.222  703d-15ad-cd02  2         Success.
2       192.168.22.3   24ff-2264-0100  2         Success.
3       192.168.22.4   24ff-2f74-0200  2         Success.
4       192.168.22.223 487a-dac8-29ba  2         Success.
```

表1-14 display smartmc vlan 命令显示信息描述表

字段	描述
ID	设备的编号
IpAddress	设备的IP地址
MacAddress	设备的MAC地址
Vlan	创建的VLAN编号
Status	创建VLAN的状态： <ul style="list-style-type: none"><li>Processing: 表示成员设备正在创建 VLAN</li><li>Success: 表示成员设备创建 VLAN 成功</li><li>Failure. The port xxx is not an access port: 成员设备因端口不是 ACCESS 口而创建 VLAN 失败</li><li>Failure. xxx not exist: 成员设备因端口不存在而创建 VLAN 失败</li></ul>

【相关命令】

- smartmc vlan

1.1.16 match

match 命令用来配置 SmartMC 组的成员设备匹配规则。

undo match 命令用来删除 SmartMC 组的成员设备匹配规则。

### 【命令】

```
match { device-type device-type | ip-address ip-address { ip-mask-length |  
ip-mask } | mac-address mac-address mac-mask-length }  
undo match { device-type device-type | ip-address ip-address  
{ ip-mask-length | ip-mask } | mac-address mac-address mac-mask-length }
```

### 【缺省情况】

未配置 SmartMC 组的成员设备匹配规则。

### 【视图】

SmartMC 组视图

### 【缺省用户角色】

network-admin

### 【参数】

**device-type device-type**: 按照设备类型匹配成员设备。

**ip-address ip-address { ip-mask-length | ip-mask }**: 按照 IP 地址匹配成员设备。

- **ip-address**: 表示成员设备的 IP 地址，为点分十进制格式。
- **ip-mask**: 成员设备的 IP 地址对应的子网掩码，为点分十进制格式。
- **ip-mask-length**: 子网掩码长度，即掩码中连续“1”的个数，取值范围为 1~32。

**mac-address mac-address mac-mask-length**: 按照 MAC 地址匹配成员设备。

- **mac-address**: 表示成员设备的 MAC 地址，格式为 H-H-H。
- **mac-mask-length**: MAC 地址掩码长度，即掩码中连续“1”的数量，取值范围为 1~48。

### 【举例】

# 创建名称为 a 的 SmartMC 组，并将 192.168.1.0/24 网段的成员设备加入该组。

```
<Sysname> system-view  
[Sysname] smartmc group a  
[Sysname-smartmc-group-a] match ip-address 192.168.1.0 24
```

### 【相关命令】

- **smartmc group**
- **display smartmc group**

## 1.1.17 smartmc auto-link-aggregation enable

**smartmc auto-link-aggregation enable** 命令用来开启以太网链路全自动聚合功能。

**undo smartmc auto-link-aggregation enable** 命令用来关闭以太网链路全自动聚合功能。

### 【命令】

```
smartmc auto-link-aggregation enable  
undo smartmc auto-link-aggregation enable
```

### 【缺省情况】

以太网链路全自动聚合功能处于关闭状态。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【使用指导】

管理设备和成员设备之间不会进行链路聚合。

开启/关闭以太网链路全自动聚合功能后，会导致网络震荡，成员设备会短时间离线。

### 【举例】

# 开启以太网链路全自动聚合功能。

```
<Sysname> system-view
[Sysname] smartmc auto-link-aggregation enable
```

## 1.1.18 smartmc auto-replace enable

**smartmc auto-replace enable** 命令用来开启自动替换故障成员设备功能。

**undo smartmc auto-replace enable** 命令用来关闭自动替换故障成员设备功能。

### 【命令】

```
smartmc auto-replace enable
undo smartmc auto-replace enable
```

### 【缺省情况】

自动替换故障成员设备功能处于关闭状态。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【使用指导】

开启自动替换故障成员设备功能前，请先将新成员设备安装到原故障成员设备的位置，并连接好线缆。

新加入的成员设备的邻居关系、产品型号、IRF 编号必须与故障成员设备一致。

### 【举例】

# 开启自动替换故障成员设备功能。

```
<Sysname> system-view
[Sysname] smartmc auto-replace enable
```

### 【相关命令】

- **smartmc replace**

### 1.1.19 smartmc backup configuration

**smartmc backup configuration** 命令用来手动备份设备的配置文件。

#### 【命令】

**smartmc backup configuration** { **group** *group-name-list* | **tc** [ *tc-id-list* ] }

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**group** *group-name-list*: 手动备份 SmartMC 组的配置文件。*group-name-list* 为 SmartMC 组名称列表, 表示方式为 *group-name-list* = { *group-name1* *group-name2* }&<1-10>, 其中 *group-name1* 和 *group-name2* 为 SmartMC 组的名称, &<1-10>表示前面的参数最多可以输入 10 次。组名称为 1~31 个字符的字符串, 区分大小写。

**tc** *tc-id-list*: 手动备份设备的配置文件。*tc-id-list* 为设备编号列表, 表示方式为 *tc-id-list* = { *tc-id1* [ **to** *tc-id2* ] }&<1-10>, *tc-id* 取值范围为 0~255。其中, *tc-id1* 和 *tc-id2* 为设备编号, *tc-id2* 必须大于等于 *tc-id1* 的设备编号, &<1-10>表示前面的参数最多可以输入 10 次。如果不指定 *tc-id-list* 参数, 则对所有设备进行备份操作。

#### 【使用指导】

配置该功能后, 设备将当前运行的配置保存到配置文件中, 并自动上传到 FTP 服务器。

备份配置文件命名格式为 *设备桥 MAC 地址\_backup.cfg*。

#### 【举例】

# 手动备份编号为 1~4 的成员设备的配置文件。

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

```
[Sysname] smartmc backup configuration tc 1 to 4
```

# 手动备份名称为 test1、test2 和 test3 的 SmartMC 组的配置文件。

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

```
[Sysname] smartmc backup configuration group test1 test2 test3
```

#### 【相关命令】

- **display smartmc configuration**
- **smartmc backup configuration interval**

### 1.1.20 smartmc backup configuration max-number

**smartmc backup configuration max-number** 命令用来配置同时备份配置文件的成员设备的最大个数。

**undo smartmc backup configuration max-number** 命令用来恢复缺省情况。

#### 【命令】

**smartmc backup configuration max-number** *max-number*

**undo smartmc backup configuration max-number**

**【缺省情况】**

同时备份配置文件的成员设备的最大个数为 5。

**【视图】**

系统视图

**【缺省用户角色】**

network-admin

**【参数】**

*max-number*: 同时备份配置文件的成员设备的最大个数，取值范围为 2~20。

**【使用指导】**

同时备份配置文件的成员设备的最大个数和 FTP 服务器性能相关。如果发现成员设备备份配置文件失败，请将同时备份配置文件的成员设备的最大个数设置为较小值。

**【举例】**

```
# 配置同时备份配置文件的成员设备的最大个数为 10。  
<Sysname> system-view  
[Sysname] smartmc backup configuration max-number 10
```

**【相关命令】**

- **display smartmc configuration**
- **smartmc backup configuration**
- **smartmc backup configuration interval**

### 1.1.21 smartmc backup configuration interval

**smartmc backup configuration interval** 命令用来开启自动备份配置文件功能，并配置自动备份配置文件的时间间隔。

**undo smartmc backup configuration interval** 命令用来关闭自动备份配置文件功能。

**【命令】**

```
smartmc backup configuration interval interval  
undo smartmc backup configuration interval
```

**【缺省情况】**

自动备份配置文件功能处于关闭状态。

**【视图】**

系统视图

**【缺省用户角色】**

network-admin

**【参数】**

*interval*: 自动备份配置文件的时间间隔，取值范围为 1~720，单位为小时。

### 【使用指导】

配置该功能后，配置文件会立即备份一次，以后根据该时间定期备份。

管理设备和成员设备按照本命令配置的时间间隔自动将当前运行配置保存到配置文件中，并自动上传到 FTP。

备份配置文件命名格式为 *设备桥 MAC 地址\_backup.cfg*。

### 【举例】

# 开启自动备份配置文件功能，并配置自动备份配置文件的时间间隔为 24 小时。

```
<Sysname> system-view
[Sysname] smartmc backup configuration interval 24
```

### 【相关命令】

- **display smartmc configuration**
- **smartmc backup configuration**

## 1.1.22 smartmc batch-file apply

**smartmc batch-file apply** 命令用来配置成员设备连接 AP 或 IP 电话时，端口使用的命令行批处理文件。

**undo smartmc batch-file apply** 命令用来取消成员设备连接 AP 或 IP 电话时，端口使用的命令行批处理文件。

### 【命令】

```
smartmc batch-file batch-file-name apply { ap | phone }
undo smartmc batch-file apply { ap | phone }
```

### 【缺省情况】

未配置成员设备连接 AP 或 IP 电话时端口使用的命令行批处理文件。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**batch-file-name**: 命令行批处理文件名称，为 1~255 个字符的字符串，不区分大小写。

**ap**: 配置成员设备连接 AP 时，端口使用的命令行批处理文件。

**phone**: 配置成员设备连接 IP 电话时，端口使用的命令行批处理文件。

### 【使用指导】

管理设备将接入 SmartMC 网络的设备分为三类：AP、IP 电话以及其它类型的设备。

配置完成后，当管理设备通过 LLDP 感知到有 AP 或者 IP 电话接入 SmartMC 网络时：

- 若存在对应设备类型的预设模板，会先将端口下的配置恢复到缺省情况，再根据设备类型，自动向端口下发指定的命令行批处理文件中的配置。
- 若不存在对应设备类型的预设模板，则保持原有配置。



当 AP 或者 IP 电话和 SmartMC 网络断开连接后，对应端口的配置不会改变，当有设备再次通过该端口接入 SmartMC 网络时：

- 如果接入的是 AP 或者 IP 电话，但是和上次接入的设备类型不同，若存在对应设备类型的预设模板，管理设备会先将端口下的配置恢复到缺省情况，再向此端口下发配置；若不存在对应设备类型的预设模板，保持原有配置。
- 如果接入的不是 AP 和 IP 电话，则保持原有配置。

#### 【举例】

# 配置成员设备连接 AP 时，端口使用的命令行批处理文件为 ap.cmdset 文件。

```
<Sysname> system-view
[Sysname] smartmc batch-file ap.cmdset apply ap
```

#### 【相关命令】

- **create batch-file**

### 1.1.23 smartmc batch-file deploy

**smartmc batch-file deploy** 命令用来向成员设备或 SmartMC 组批量下发命令行。

#### 【命令】

```
smartmc batch-file batch-file-name deploy { group group-name-list | tc tc-id-list }
```

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*batch-file-name*：命令行批处理文件名称，为 1~255 个字符的字符串。

**group** *group-name-list*：将命令行批量下发到指定的 SmartMC 组。*group-name-list* 为 SmartMC 组名称列表，表示方式为 *group-name-list* = { *group-name1* *group-name2* }&<1-10>，其中 *group-name1* 和 *group-name2* 为 SmartMC 组的名称，&<1-10> 表示前面的参数最多可以输入 10 次。组名称为 1~31 个字符的字符串，区分大小写。

**tc** *tc-id-list*：将命令行批量下发到指定的成员设备。*tc-id-list* 为成员设备编号列表，表示方式为 *tc-id-list* = { *tc-id1* [ **to** *tc-id2* ] }&<1-10>，*tc-id* 取值范围为 1~255。其中，*tc-id1* 和 *tc-id2* 为成员设备编号，*tc-id2* 必须大于等于 *tc-id1* 的成员设备编号，&<1-10> 表示前面的参数最多可以输入 10 次。

#### 【举例】

# 向 SmartMC 组 testgroup 批量下发名称为 startup.cmdset 的命令行批处理文件。

```
<Sysname> system-view
[Sysname] smartmc batch-file startup.cmdset deploy group testgroup
```

#### 【相关命令】

- **create batch-file**

- `display smartmc batch-file status`

#### 1.1.24 smartmc enable

`smartmc enable` 命令用来开启 SmartMC 功能并配置设备的角色。

`undo smartmc enable` 命令用来关闭 SmartMC 功能。

##### 【命令】

```
smartmc { tc | tm username username password { cipher | simple } string }  
enable  
undo smartmc enable
```

##### 【缺省情况】

SmartMC 功能处于关闭状态。

##### 【视图】

系统视图

##### 【缺省用户角色】

network-admin

##### 【参数】

**tc**: 开启 SmartMC 功能并配置设备的角色为成员设备。

**tm**: 开启 SmartMC 功能并配置设备的角色为管理设备。

**username username**: 本地用户的用户名，为 1~55 个字符的字符串，区分大小写。

**password**: 本地用户的密码。

**cipher**: 表示以密文方式设置的密码。

**simple**: 表示以明文方式设置的密码，该密码将以密文形式存储。

**string**: 密码字符串，区分大小写。明文密码为 1~63 个字符的字符串，密文密码为 1~117 个字符的字符串。

##### 【使用指导】

一个 SmartMC 网络有且仅能有一台管理设备。

管理设备和成员设备都需要配置该命令，其他 SmartMC 命令只能在管理设备上配置。

将管理设备切换为成员设备或关闭 SmartMC 功能时，会清空当前运行配置中与 SmartMC 功能相关配置。

SmartMC 功能会占用一定的 ACL 资源，如果 ACL 资源不足，会导致 SmartMC 功能开启失败。请使用 `display acl` 命令查看 ACL 的配置和运行情况，并根据实际情况使用 `undo acl` 命令删除不需要的 ACL。释放 ACL 资源后，再开启 SmartMC 功能。有关 ACL 的详细介绍，请参见“ACL 和 QoS 配置指导”中的“ACL”。

开启 SmartMC 功能时，设备会检查 80 端口、443 端口是否被占用，因为 HTTP 和 HTTPS 服务需要占用这两个端口号，如果 80 端口或者 443 端口被占用，则 SmartMC 功能开启失败。

##### 【举例】

# 开启 SmartMC 功能并配置设备的角色为成员设备。

```
<Sysname> system-view
[Sysname] smartmc tc enable
```

### 1.1.25 smartmc ftp-server

**smartmc ftp-server** 命令用来配置 FTP 服务器信息。

**undo smartmc ftp-server** 命令用来恢复缺省情况。

#### 【命令】

```
smartmc ftp-server server-address username username password { cipher |
simple } string
undo smartmc ftp-server
```

#### 【缺省情况】

未配置 FTP 服务器信息。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**server-address**: FTP 服务器的 IP 地址。

**username username**: FTP 服务器的用户名，为 1~55 个字符的字符串，区分大小写。

**password**: FTP 服务器的密码。

**cipher**: 表示以密文方式设置的密码。

**simple**: 表示以明文方式设置的密码，该密码将以密文形式存储。

**string**: 密码字符串，区分大小写。明文密码为 1~63 个字符的字符串，密文密码为 1~117 个字符的字符串。

#### 【使用指导】

多次执行该命令，最后一次执行的命令生效。

#### 【举例】

# 配置 FTP 服务器的 IP 地址为 192.168.22.19，用户名和密码为 admin。

```
<Sysname> system-view
[Sysname] smartmc ftp-server 192.168.22.19 username admin password simple admin
```

#### 【相关命令】

- **display smartmc configuration**

### 1.1.26 smartmc group

**smartmc group** 命令用来创建 SmartMC 组，并进入 SmartMC 视图。如果指定的 SmartMC 组已经存在，则直接进入 SmartMC 组视图。

**undo smartmc group** 命令用来删除 SmartMC 组。

### 【命令】

```
smartmc group group-name  
undo smartmc group group-name
```

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*group-name*: SmartMC 组的名称，为 1~31 个字符的字符串，区分大小写。

### 【使用指导】

创建 SmartMC 组并将设备加入 SmartMC 组后，对设备进行配置或升级时，指定对应的 SmartMC 组即可完成对组内所有成员设备的操作。

### 【举例】

```
# 创建 SmartMC 组 testgroup。  
<Sysname> system-view  
[Sysname] smartmc group testgroup  
[Sysname-smartmc-group-testgroup]
```

### 【相关命令】

- match

## 1.1.27 smartmc outbound

**smartmc outbound** 命令用来将接口配置为 SmartMC 网络的出接口。

**undo smartmc outbound** 命令用来恢复为缺省情况。

### 【命令】

```
smartmc outbound  
undo smartmc outbound
```

### 【缺省情况】

未将接口配置为 SmartMC 网络的出接口，SmartMC 网络无法和外部网络互通。

### 【视图】

三层以太网接口视图  
VLAN 接口视图

### 【缺省用户角色】

network-admin

### 【使用指导】

SmartMC 网络在 VLAN1 内建立，不能将 Vlan-interface1 配置为 SmartMC 网络的出接口。

### 【举例】

# 将接口 GigabitEthernet1/0/1 配置为 SmartMC 网络的出接口。

```
<Sysname> system-view
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] smartmc outbound
```

## 1.1.28 smartmc resource-monitor

**smartmc resource-monitor** 命令用来开启资源监控功能。

**undo smartmc resource-monitor** 命令用来关闭资源监控功能。

### 【命令】

```
smartmc resource-monitor [ cpu | memory | temperature ] * [ group
group-name-list | tc { tc-id-list | mac-address mac-address } | tm ]
undo smartmc resource-monitor [ cpu | memory | temperature ] * [ group
group-name-list | tc { tc-id-list | mac-address mac-address } | tm ]
```

### 【缺省情况】

资源监控功能处于关闭状态。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**cpu**: 开启对 CPU 使用率的监控。

**memory**: 开启对内存使用情况的监控。

**temperature**: 开启对设备温度的监控。

**group group-name-list**: 开启 SmartMC 组的资源监控功能, *group-name-list* 为 SmartMC 组名称列表, 表示方式为 *group-name-list* = { *group-name1* *group-name2* } &<1-10>, 其中 *group-name1* 和 *group-name2* 为 SmartMC 组的名称, &<1-10> 表示前面的参数最多可以输入 10 次。组名称为 1~31 个字符的字符串, 区分大小写。

**tc**: 开启成员设备的资源监控功能。

**tc-id-list**: 成员设备编号。 *tc-id-list* = { *tc-id1* [ **to** *tc-id2* ] } &<1-10>, *tc-id* 取值范围为 1~255。其中, *tc-id1* 和 *tc-id2* 为成员设备编号, *tc-id2* 必须大于等于 *tc-id1* 的成员设备编号, &<1-10> 表示前面的参数最多可以输入 10 次。

**mac-address mac-address**: 开启 MAC 地址对应成员设备的资源监控功能。 *mac-address* 为成员设备的 MAC 地址, 格式为 H-H-H。

**tm**: 开启管理设备的资源监控功能。

### 【使用指导】

如果不指定资源类型, 则开启成员设备上所有资源的监控功能。

如果不指定设备的角色, 则开启管理设备和所有成员设备的资源监控功能。

### 【举例】

# 开启编号为 1、2 和 3 的成员设备上所有资源的监控功能。

```
<Sysname> system-view
[Sysname] smartmc resource-monitor tc 1 to 3
```

### 【相关命令】

- **display smartmc resource-monitor**
- **smartmc resource-monitor interval**
- **smartmc resource-monitor max-age**

## 1.1.29 smartmc resource-monitor interval

**smartmc resource-monitor interval** 命令用来配置管理设备获取成员设备资源监控数据的时间间隔。

**undo smartmc resource-monitor interval** 命令用来恢复缺省情况

### 【命令】

```
smartmc resource-monitor interval interval
undo smartmc resource-monitor interval
```

### 【缺省情况】

管理设备获取成员设备资源监控数据的时间间隔为 1 分钟。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**interval**: 管理设备获取成员设备资源监控数据的时间间隔，取值范围为 1~60，单位为分钟。

### 【举例】

# 配置管理设备获取成员设备资源监控数据的时间间隔为 5 分钟。

```
<Sysname> system-view
[Sysname] smartmc resource-monitor interval 5
```

### 【相关命令】

- **display smartmc resource-monitor**
- **smartmc resource-monitor**

## 1.1.30 smartmc resource-monitor max-age

**smartmc resource-monitor max-age** 命令用来配置资源监控数据的老化时间。

**undo smartmc resource-monitor max-age** 命令用来恢复缺省情况。

### 【命令】

```
smartmc resource-monitor max-age max-age
```

**undo smartmc resource-monitor max-age**

**【缺省情况】**

资源监控数据的老化时间为 24 小时。

**【视图】**

系统视图

**【缺省用户角色】**

network-admin

**【参数】**

**max-age**: 资源监控数据的老化时间，取值范围为 1～168，单位为小时。

**【使用指导】**

当资源监控数据老化时间到达后，管理设备重新收集各设备的资源监控数据。

**【举例】**

# 配置资源监控数据的老化时间为 1 小时。

```
<Sysname> system-view
[Sysname] smartmc resource-monitor max-age 1
```

**【相关命令】**

- **display smartmc resource-monitor**
- **smartmc resource-monitor**

### 1.1.31 smartmc replace

**smartmc replace** 命令用来手动替换故障成员设备。

**【命令】**

**smartmc replace tc tc-id1 faulty-tc tc-id2**

**【视图】**

系统视图

**【缺省用户角色】**

network-admin

**【参数】**

**tc tc-id1**: 指定新加入的成员设备，**tc-id1** 为新加入的成员设备的编号，取值范围为 1～255。

**faulty-tc tc-id2**: 指定故障成员设备，**tc-id1** 为故障成员设备的编号，取值范围为 1～255。

**【使用指导】**

替换故障成员设备前，请先将新成员设备安装到原故障成员设备的位置，并连接好线缆。

新加入的成员设备的邻居关系、产品型号、IRF 编号必须与故障成员设备一致。

**【举例】**

# 使用编号为 10 的成员设备替换编号为 5 的故障成员设备。

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

```
[Sysname] smartmc replace tc 10 faulty-tc 5
```

#### 【相关命令】

- **display smartmc replace status**
- **smartmc auto-replace enable**

### 1.1.32 smartmc tc boot-loader

**smartmc tc boot-loader** 命令用来配置升级成员设备使用的启动软件。

**undo smartmc tc boot-loader** 命令用来取消升级成员设备使用的启动软件。

#### 【命令】

```
smartmc tc tc-id boot-loader { ipe-filename | boot boot-filename system  
system-filename }  
undo smartmc tc tc-id boot-loader
```

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**tc tc-id**: 需要配置升级文件的成员设备编号，取值范围为 1~255。

**ipe-filename**: 指定成员设备升级使用的 IPE 文件，为 5~45 个字符的字符串，不区分大小写，以.ipe 为后缀名。

**boot boot-filename**: 指定成员设备升级使用的 Boot 文件，为 5~45 个字符的字符串，不区分大小写，以.bin 为后缀。

**system system-filename**: 指定成员设备升级使用的 System 文件，为 5~45 个字符的字符串，不区分大小写，以.bin 为后缀。

#### 【举例】

# 指定编号为 1 的升级成员设备使用的启动软件包为 boot.bin 和 system.bin 文件。

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

```
[Sysname] smartmc tc 1 boot-loader boot boot.bin system system.bin
```

#### 【相关命令】

- **display smartmc tc**

### 1.1.33 smartmc tc device-type

**smartmc tc device-type** 命令用来自定义成员设备的类型。

**undo smartmc tc device-type** 命令用来删除自定义的成员设备类型。

#### 【命令】

```
smartmc tc sysoid sysoid device-type device-type  
undo smartmc tc sysoid sysoid device-type device-type
```



### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**sysoid** *sysoid*: 设备的 SYSOID。

**device-type** *device-type*: 指定成员设备的类型。

### 【使用指导】

用户可以执行 **display smartmc tc verbose** 命令获取成员设备的 SYSOID。

### 【举例】

# 自定义 SYSOID 为 1.3.6.1.4.1.25506.1.1588, device-type 为 SW 的设备类型。

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

```
[Sysname] smartmc tc sysoid 1.3.6.1.4.1.25506.1.1588 device-type SW
```

## 1.1.34 smartmc tc password

**smartmc tc password** 命令用来修改成员设备缺省用户的密码。

### 【命令】

```
smartmc tc password string
```

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*string*: 密码字符串, 为 1~63 个字符的字符串, 区分大小写。

### 【使用指导】

管理设备和成员设备建立连接时, 使用缺省用户名 (admin) 和密码 (admin) 与成员设备建立 NETCONF 会话, 并将成员设备加入到 SmartMC 网络中。SmartMC 网络组建完成后, 建议用户修改 admin 用户的密码, 提高 SmartMC 网络的安全性。

该功能只能修改自动加入 SmartMC 网络的成员设备。

### 【举例】

# 修改成员设备缺省用户的密码为 Admin123

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

```
[Sysname] smartmc tc password Admin123
```

## 1.1.35 smartmc tc startup-configuration

**smartmc tc startup-configuration** 命令用来配置升级成员设备使用的配置文件。

**undo smartmc tc startup-configuration** 命令用来取消升级成员设备使用的配置文件。

### 【命令】

```
smartmc tc tc-id startup-configuration cfg-filename  
undo smartmc tc tc-id startup-configuration
```

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**tc tc-id**: 升级配置文件的成员设备编号，取值范围为 1~255。

**cfg-filename**: 配置文件的名称，为 5~45 个字符的字符串，不区分大小写，以.cfg 为后缀。

### 【举例】

# 指定编号为 1 的升级成员设备使用的配置文件为 startup.cfg。

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

```
[Sysname] smartmc tc 1 startup-configuration startup.cfg
```

### 【相关命令】

- **display smartmc tc**

## 1.1.36 smartmc topology-refresh

**smartmc topology-refresh** 命令用来手动刷新网络拓扑。

### 【命令】

```
smartmc topology-refresh
```

### 【视图】

任意视图

### 【缺省用户角色】

network-admin

### 【使用指导】

网络拓扑发生变化时，需要先刷新网络拓扑，再查看网络的最新拓扑。

### 【举例】

# 手动刷新网络拓扑。

```
<Sysname> smartmc topology-refresh
```

### 【相关命令】

- **display smartmc device-link**

## 1.1.37 smartmc topology-refresh interval

**smartmc topology-refresh interval** 命令用来配置自动刷新网络拓扑的时间间隔。

**undo smartmc topology-refresh interval** 命令用来恢复缺省情况。

#### 【命令】

```
smartmc topology-refresh interval interval
undo smartmc topology-refresh interval
```

#### 【缺省情况】

自动刷新网络拓扑的时间间隔为 60 秒。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*interval*: 自动刷新网络拓扑的时间间隔，取值范围为 60～300，单位为秒。

#### 【举例】

```
# 配置自动刷新网络拓扑的时间间隔为 100 秒。
<Sysname> system-view
[Sysname] smartmc topology-refresh interval 100
```

#### 【相关命令】

- **display smartmc device-link**

### 1.1.38 smartmc topology-save

**smartmc topology-save** 命令用来保存当前拓扑。

#### 【命令】

```
smartmc topology-save
```

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【使用指导】

执行该命令将当前网络拓扑结构保存到 Flash 中的拓扑文件 topology.db 中。当管理设备重启或由成员设备重切换为管理设备后，管理设备和成员设备根据该拓扑文件恢复原来的网络拓扑结构。

#### 【举例】

```
# 保存当前网络拓扑。
<Sysname> system-view
[Sysname] smartmc topology-save
```

#### 【相关命令】

- **display smartmc device-link**

### 1.1.39 smartmc upgrade boot-loader

**smartmc upgrade boot-loader** 命令用来升级成员设备或 SmartMC 组的启动软件。

**undo smartmc upgrade** 命令用来取消延迟升级或定时升级。

#### 【命令】

```
smartmc upgrade boot-loader { group | tc } list [ delay minutes | time time ]
smartmc upgrade boot-loader { group | tc } { list { boot boot-filename system
system-filename | file ipe-filename } }<1-40> [ delay delay-time | time
time ]
undo smartmc upgrade
```

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**group**: 表示升级 SmartMC 组的启动软件。

**tc**: 表示升级成员设备的启动软件。

**list**: SmartMC 组名称列表或成员设备编号列表:

- 指定 SmartMC 组名称列表时, *list* = { *group-name1* *group-name2* }, 其中 *group-name1* 和 *group-name2* 为 SmartMC 组的名称。组名称为 1~31 个字符的字符串, 区分大小写。
- 指定成员设备编号列表时, *list* = { *tc-id1* [ **to** *tc-id2* ] }, *tc-id* 取值范围为 1~255。其中, *tc-id1* 和 *tc-id2* 为成员设备编号, *tc-id2* 必须大于等于 *tc-id1* 的成员设备编号。

**boot** *boot-filename*: Boot 包, *boot-filename* 为 Boot 包的名称。

**system** *system-filename*: System 包, *system-filename* 为 System 包的名称。

**file** *ipe-filename*: IPE (Image Package Envelope, 复合软件包套件) 文件, *ipe-filename* 为 IPE 文件的名称, 为 5~45 个字符的字符串, 不区分大小写, 以 *ipe* 为后缀名。

**delay** *delay-time*: 表示延时升级, 延迟时长为 *delay-time*, 取值范围为 1~1440, 单位为分钟。

**time** *in-time*: 表示定时升级, 格式为 HH:MM (小时:分钟)。HH 取值范围为 0~23, MM 取值范围为 0~59。

#### 【使用指导】

升级成员设备的启动软件时, 若不指定启动软件的名称, 需要先通过 **smartmc tc boot-loader** 命令指定成员设备使用的启动软件。

升级 SmartMC 组的启动软件时, 若不指定启动软件的名称, 需要先进入 SmartMC 组视图通过 **boot-loader** 命令指定该组使用的启动软件。

设备同时仅能够进行一个升级操作, 请完成一次升级操作后, 再进行下一个升级操作。

如果不指定 **delay** 和 **time** 参数，表示立即升级成员设备或 SmartMC 组的启动软件。如果选择了立即升级启动软件，则无法取消升级；如果选择了延时升级或者定时升级启动软件，在设备开始升级前，还可以通过 **undo smartmc upgrade** 命令取消升级。

#### 【举例】

# 升级 SmartMC 组 test1 和 test2 中成员设备的 boot.bin 文件和 sys.bin 文件。

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

```
[Sysname] smartmc upgrade boot-loader group test1 test2 boot boot.bin system sys.bin
```

#### 【相关命令】

- **boot-loader**
- **startup-configuration**

### 1.1.40 smartmc upgrade startup-configuration

**smartmc upgrade startup-configuration** 命令用来升级成员设备或 SmartMC 组的配置文件。

**undo smartmc upgrade** 命令用来取消延迟升级或定时升级。

#### 【命令】

```
smartmc upgrade startup-configuration { group | tc } list [ delay minutes  
| time time ]  
smartmc upgrade startup-configuration { group | tc } { list  
cfg-filename }&lt;1-40> [ delay delay-time | time time ]  
undo smartmc upgrade
```

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**group**: 表示升级 SmartMC 组的配置文件。

**tc**: 表示升级成员设备的配置文件。

**list**: SmartMC 组名称列表或成员设备编号列表:

- 指定 SmartMC 组名称列表时，**list** = { *group-name1* *group-name2* }，其中 *group-name1* 和 *group-name2* 为 SmartMC 组的名称。组名称为 1~31 个字符的字符串，区分大小写。
- 指定成员设备编号列表时，**list** = { *tc-id1* [ **to** *tc-id2* ] }，*tc-id* 取值范围为 1~255。其中，*tc-id1* 和 *tc-id2* 为成员设备编号，*tc-id2* 必须大于等于 *tc-id1* 的成员设备编号。

**cfg-filename**: 配置文件的名称。

**delay delay-time**: 表示延时升级，延迟时长为 *delay-time*，取值范围为 1~1440，单位为分钟。

**time time**: 表示定时升级, 格式为 HH:MM (小时:分钟)。HH 取值范围为 0~23, MM 取值范围为 0~59。

#### 【使用指导】

升级成员设备的配置文件时, 若不指定配置文件的名称, 需要先通过 **smartmc tc startup-configuration** 命令指定成员设备使用的配置文件。

升级 SmartMC 组的配置文件时, 若不指定配置文件的名称, 需要先进入 SmartMC 组视图通过 **startup-configuration** 命令指定该组使用的配置文件。

配置文件升级过程中, 设备不需要重启。

设备同时仅能够进行一个升级操作, 请完成一次升级操作后, 再进行下一个升级操作。

如果不指定 **delay** 和 **time** 参数, 表示立即升级成员设备或 SmartMC 组的配置文件。如果选择了立即升级配置文件, 则无法取消升级; 如果选择了延时升级或者定时升级配置文件, 在设备开始升级前, 还可以通过 **undo smartmc upgrade** 命令取消升级。

#### 【举例】

# 升级 SmartMC 组 test1 和 test2 中的配置文件 startup.cfg。

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

```
[Sysname] smartmc upgrade startup-configuration group test1 test2 startup.cfg
```

#### 【相关命令】

- **boot-loader**
- **startup-configuration**

### 1.1.41 smartmc vlan

**smartmc vlan** 命令用来为成员设备创建 VLAN。

#### 【命令】

```
smartmc vlan vlan-id { group group-name-list | tc tc-id-list }
```

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**vlan-id**: VLAN 的编号, 取值范围为 1~4094。

**group** *group-name-list*: 为 SmartMC 组创建 VLAN, *group-name-list* 为 SmartMC 组名称列表, 最多可以输入 10 个 SmartMC 组名。组名称为 1~31 个字符的字符串, 区分大小写。

**tc** *tc-id-list*: 为成员设备创建 VLAN。*tc-id-list* = { *tc-id1* [ **to** *tc-id2* ] } &<1-10>, *tc-id* 取值范围为 1~255。其中, *tc-id1* 和 *tc-id2* 为成员设备编号, *tc-id2* 必须大于等于 *tc-id1* 的成员设备编号, &<1-10> 表示前面的参数最多可以输入 10 次。

#### 【使用指导】

请保证在网络拓扑稳定的情况下配置该命令。执行该命令前, 建议用户先使用 **smartmc topology-refresh** 命令刷新网络拓扑。

为成员设备创建 VLAN 后，会将 Access 类型端口加入 VLAN。

对于如下情况，端口不会加入创建的 VLAN 中：

- 成员设备上与管理设备相连的 Access 类型端口。
- 成员设备之间直接相连的 Access 类型端口。
- 连接离线设备的 Access 类型端口。需要用户手动删除离线设备后再次执行该命令。

如果成员设备创建 VLAN 成功，但是向 VLAN 中添加端口失败，则该成员设备的 VLAN 相关配置将恢复到创建前的状态。

一台成员设备创建 VLAN 失败不会影响其他成员设备的 VLAN 创建。

配置该命令后，请使用 **display smartmc vlan** 命令查看成员设备是否成功创建 VLAN。

#### 【举例】

# 为成员设备 1 和成员设备 2 创建 VLAN 2。

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

```
[Sysname] smartmc vlan 2 tc 1 to 2
```

As a best practice, execute the display smartmc vlan command to verify that the VLAN has been created successfully.

### 1.1.42 startup-configuration

**startup-configuration** 命令用来配置 SmartMC 组的配置文件。

**undo startup-configuration** 命令用来恢复缺省情况。

#### 【命令】

**startup-configuration** *cfgfile*

**undo startup-configuration**

#### 【缺省情况】

未配置 SmartMC 组的配置文件。

#### 【视图】

SmartMC 组视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*cfgfile*: 配置文件的名称，为 5~45 个字符的字符串，文件名后缀必须为.cfg。

#### 【使用指导】

多次执行该命令，最后一次执行的配置生效。

#### 【举例】

# 配置 SmartMC 组 testgroup 的配置文件为 startup.cfg。

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

```
[Sysname] smartmc group testgroup
```

```
[Sysname-smartmc-group-testgroup] startup-configuration startup.cfg
```

### 【相关命令】

- `smartmc group`