思科 IOS IP SLA 详解

李明(juechen70)　编译

注：如需要 Word 版本请与本人联系

思科 IOS SLA(以下简称 IP SLA)基于思科 SAA 技术发展而来，并在其基础上进行了增强，该特性让用户可以监测路两台思科路由器之间或思科路由器与一个远程的 IP 设备之间的网络性能。本文集中讨论了新的 SLA（又称作 SAA）信息，包括使用与配置原则，如何从 IP SLA 中抽取数据，如何使用命令行(CLI)和 SNMP 来配置 IP SLA 等内容。SNMP 的 MIB 细节可以参考 Cisco-RTTMON-MIB 。

1 IP SLA 概述

1.1 IP SLA 用途

IP SLA 可以用于以下用途

- SLA 监测
- 网络性能监测
- 网络服务评估
- 端到端的可用性监测
- 网络故障诊断
- MPLS 网络监测
- VOIP 网络监测

1.2 IP SLA 优点

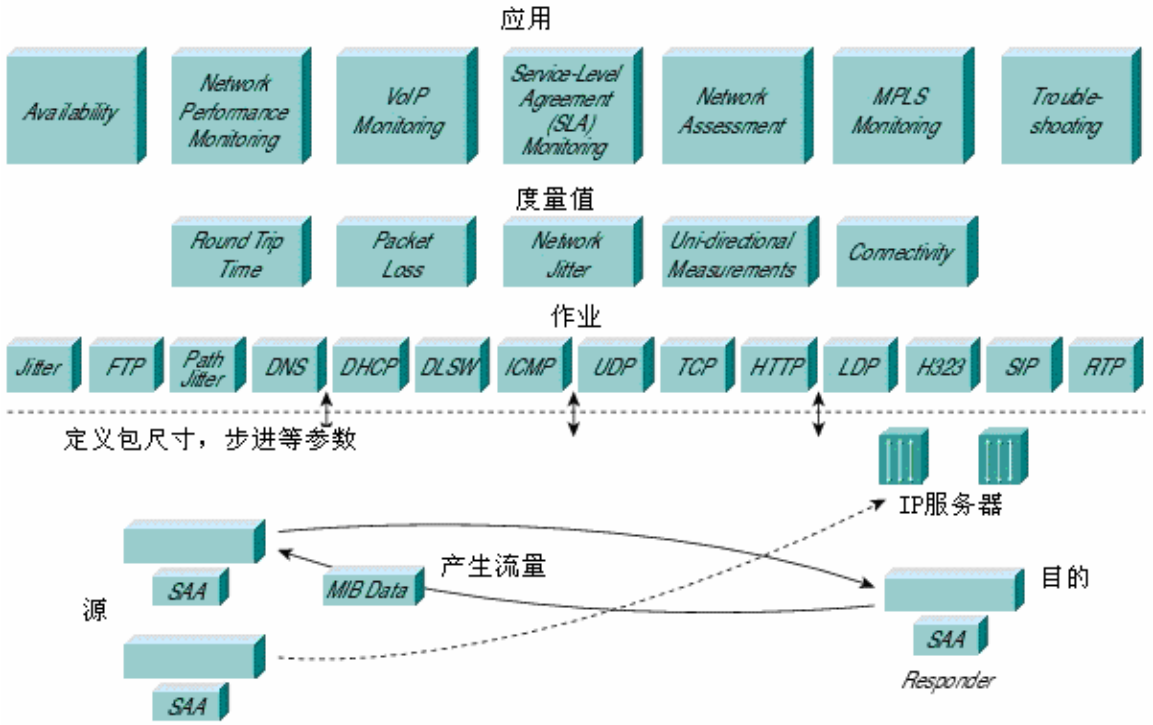
- 增强布署新应用的信心
- 监测与确认服务质量，实现差别服务
- 增加用户信心与用户满意度
- 通过 SLA 的度量，用户可以确认他的网络应用照他们需要的那样运行
- 网络有问题时可以预告提醒用户
- 可以连续的、可靠的周期性度量网络的性能

1.3 IP SLA 特性概述

测量能力：可以测量 UDP 响应时间、单向延时、抖动、掉包情况和连通性；ICMP 响应时间与连通性、每一跳的 ICMP 响应时间与抖动；DNS 查询、TCP 连接、HTTP 处理时间等的性能度量；丢包统计；DHCP 响应时间测试；从网络设备到服务器的响应时间；模拟 Voip 的 codec's 测试出语音质量的 MOS/ICPIF 得分；DLSw+通道性能；

IP SLA 可以通过命令行或 SNMP(Cisco-RTTMON-MIB)来实现前期告警，可以定义 SLA 的监测阈值，当一个 SLA 达到阈值时能产生一个 SNMP TRAP，并且能触发相关的作业以实现更详细的分析；IP SLA 可以实现灵活的调度，可以在任何给定的时间或以任意的时间间隔周期性运行。

图 1 Cisco IOS IP SLA



1.4 使用 IP SLA 来度量网络

IP SLA 是一个网络性能度量与诊断工具，它通过主动在多个站点之间或多条路径之间发送数量来实现对网络性能的度量。IP SLA 使用时标来计算如抖动、延时、响应时间这、丢包率、语音 MOS 值等网络性能参数。用户通过命令行或 RTTMON MIB 可以定义一个 IP SLA 动作（探针），定义 IP SLA 动作时可以明确这个动作所产生的流量的包尺寸、发包间隔、协议类型、DSCP 标记以及其它一些参数；然后让这个动作在适当的时候运行并返回度量性能所需要的参数。例如，我们可以定义一个用来度量 UDP 的抖动的动作，这个动作每分钟每隔 20ms 发出 10 个 64Bytes 的包。

由 IP SLA 探针所返回的数据存储在 RTTMON MIB 中，可以通过命令行或网络管理应用来提取其中的网络性能统计数据。可以让 IP SLA 探针在任意时间点或以任意的时间间隔连运行，通过设置不同的 DSCP 值，IP SLA 可以测试同一链路上不同类型的流量的性能。

运行 Cisco IOS 的目标路由器可以需要配置成 IP SLA Responder(响应器)，用于处理测试包并提供更详细的时标信息。响应器可以将目标路由器的处理延时等信息发送回源路由器，这样在后期计算时这个处理延时就可以去掉以提高测试的精确性。使用 IP SLA 时单向测量机制也是可行的。

IP SLA 利用 SNMP trap 提供了预告警告机制，每个测试探针可以预告配置一个性能阈值；当结果超出阈值时，IP SLA 将产生一个 SNMP TRAP 并发送到网管应用上，可用的 SNMP Trap 包括：循环时间、平均抖动、单向延时、抖动、丢包率、MOS 值、连通性测试等，同时管理员也可以定义 IP SLA 去执行一个新的探针，例如当延时超出阈值时可以触发第二个作业－－测试每一跳的延时来将问题区域隔离出来。

2 SLA 监测介绍

2.1 概述

企业 IT 部门向内部客户或其它部门提供 SLA 的压力在不断的增长，如何去度量外部提供的 SLA 也是一个大问题；服务供应商为了提高客户满意度也有提供 SLA 的动机；管理上需要确保网络能满足生产经营活动的需要；终端用户需要保证在他需要时，关键应用与服务总是可用的；一个组织在布署一个新的技术、新的关键应用或 IP 服务（如 voip），通常需要一个 SLA 或服务水平的验证。交付 SLA 的复杂度日益增加，很难决定要去监测些什么，怎么去度量，以什么样的频率去搜集数据。在一个复杂的多服务网络中，要实现端到端的服务监测也是相当困难的。服务水平管理，包括 SLA，是解决这个问题的一个关键组件，并且可以增加网络可用性。

2.2 定义 SLA 需求

在定义 SLAs 时，最为关键的是客户所关注的事务处理目标，只有在明确客户关注什么的情况下，才能制定出切实可行的服务水平参数来。太多的、太复杂的不可行的参数往往会达不到管理服务水平的目标，并受到客户的责备。一份 SLA 越通用，SLA 的需求也就越简单，服务供应商的 SLA 必须同时考虑连接性与主机应用：

- SLA 的度量是需要的
- 能够以合适的粒度去界定 SLA 问题

- Ø 出现 SLA 违背时能施以财务惩罚
- Ø 可通过 WEB 访问的商务报告和技术细节报告。
- Ø 在布署新的应用时确定关键的业务目标已达到

2.3 服务水平合同与 SLAs

SLA 是服务水平合同(SLC)中的关键组成部分，SLC 定义了服务提供者向终端用户提供的连接性与性能协议，服务提供者可以在企业内部(如企业内的 IT 部门就是其它部门的服务提供者),也可以是一家外部公司(如提供广域连接、主机服务的的 ISP)。一个 SLC 通常包括多个 SLA，所以一个特定的 SLA 的违反会引起整个 SLC 的违反。服务水平管理解决方案需要提供一个能管理一个 SLC 内所有 SLA 的解决办法。应当能独立的监测多个 SLC 及其相关的 SLA，这通常包括了预期的服务水平和最小服务水平两个参数,例如,一个为分支机构和总部提供连接的 SLC 定义了一条 64kbps 的链路，要求在一个月的计算期内，链路的平均延时在 100ms 以面，这个平均延时就是预期的服务水平，这时的最小服务水平可以是一天的计算期内，平均延时在 300ms 以内。端到端的 SLC 常用于性能监测也失效管理，IP SLA 提供了每一个 SLA 的细节信息。

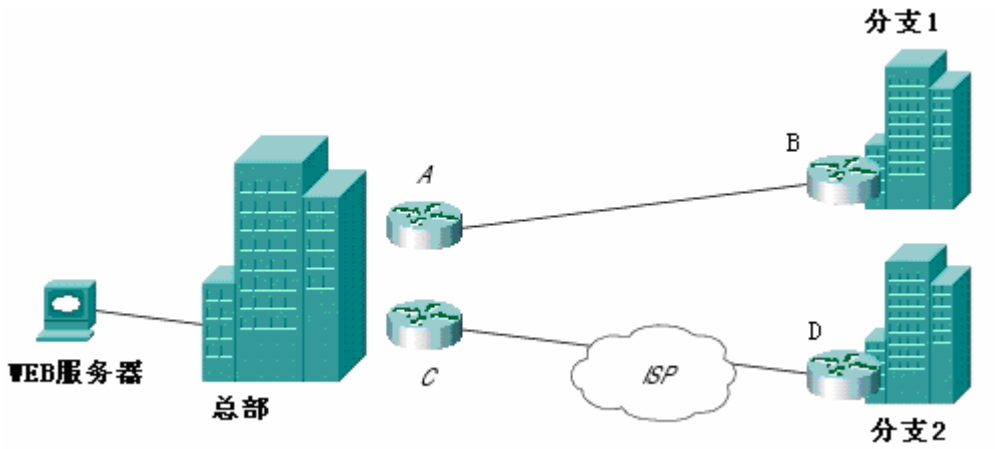
3 使用 IP SLA 监测网络

用户必须决定何时去监测服务水平的参数值，这个过程中的一个重要影响因素是所考虑的 SLA 的类型，在决定何时监测服务水平之间需要解决以下几个问题。

首先，用户必须知道服务水平合同什么时候真正开始生效？度量的主要目标是什么？什么参数是需要监测的重要参数？第二步要做的就是对网络中的流量模式进行评估，度量采样做得越多，所取得的流量模式就越准确。更多的点意味着信息更精确，相反，更多的度量采样潜在的引起管理流量的上午，让可用带宽变小。活动的度量将模拟网络中的流量型号，例如正确的包尺寸，间隔等。

3.1 例 1：测量从分支机构到中心办公室的数据流量性能

图 2 例 1 的网络拓朴



一个企业客户有一个总部，和两个分支机构办公室；其中一个分支机构使用单独的帧中继电路(256Kbps)，另一个通过 Internet VPN 的方式访问中心办公室。两个分支办公　室的用户都需要访问位于总部的 web 服务。例如，公司要求提供 99.95%的可用性，响应时间要求小于 50ms，对于通过 Internet VPN 访问总部的分支机构，提供了一个延时低于 100ms 的 SLA。基于这些数据，该企业必须考虑如何度量与验证两个分支办公室都能取开满意的服务水平，如果不能达到这个服务水平，就需要确认是网络中哪个/哪些部件(如广域链路、客户端应用、Web 服务器)导致了问题的产生。

3.1.1 选择适当的作业

在 SLA 布署中的第一步就是分析需要监测哪些参数，在前面介绍过 IP SLA 所能提供的作业类型，下面对这些作业再作详细的说明：

3.1.1.1 UDP Echo 作业

UDP ECHO 作业可以测试从路由器到 IP 设备之间的端到端的响应时间或连通性，UDP 是一个可以报告错误并提供与 IP 包处理相关的其它信息的网络层(第三层)协议，响应时间是通过计算发出 UDP ECHO 消息和收到 UDP echo 响应之间的时间差来实现的。在目标路由器上启用 IP SLA Responder 可以提高 UDP echo 的精度。在本文后面将详细讨论 IP SLA Responder。

3.1.1.2 DNS

DNS 响应时间是通过计算发出 DNS 请求和接收到响应之间的时间差来实现的，这个作业可通过用户提供的主机名查询 IP，也可通过用户提供的 IP 地址查询相应的主机名。

3.1.1.3 DHCP，动态主机配置协议

动态主机配置协议（DHCP）响应时间是计算从发出请求到收到响应之间的差值实现的，当取得 IP 地址后，源路由器会将这个 IP 释放回地址池，从而避免出现消耗大量 IP 地址的情况出现。

3.1.1.4HTTP

HTTP 作业可以度量连接或从 HTTP 服务器访问数据的往返时间(RTT)，台要通过一个 URL 来定义，HTTP 服务器响应时间度量由以下三部分组成：

- DNS 查询—域名查询的往返时间 RTT taken to perform domain name lookup
- TCP 连接—TCP 连接处理的往返时间
- HTTP 处理时间发出请求到从服务器收到响应的往返时间

3.1.2 选择适当的测试组合

在定义一个 SLA 的过程中，最困难的就是选择适当的测试组合，在做这一步之前，必须先满足以下条件：

- 源设备必须是思科设备，并且能运行 IP SLA－－IOS 版本为 12.0(5)T 或更新；
- 当做 IP SLA 作业时，目的设备可以是一个 IP 设备，使用思科路由器中的 IP SLA Responder 可以提高测量精度

当这些条件满足时，就可以集中精力选择设备组合了。一般情况下，源设备是边缘的路由器或企业网络与供应商网络的边缘路由器。

表 1：本例中所选择的测试组合：

源	目的	作业	说明
C	D	UDP	
A	B	UDP	
B	Web 服务器	HTTP	
D	Web 服务器	HTTP	
A	Web 服务器	HTTP	可选
C	Web 服务器	HTTP	可选

以上就提供了广域连接相关的细节，如 DNS 查询时间、TCP 连接及最终的 HTTP 作业等。

3.1.3 选择适当的荷载

荷载是包所承载内容的实际尺寸，这个值与包本身的大小不一样；根据协议类型的不同，包头的长度也不同。在选择荷载值时，必须考虑控制包分片的最大传输单元(MTU)的影响，通过控制与 MTU 相关的荷载尺寸，就可以控制每个采样所发出的包的个数。这个值最好能和网络中实际包的值相一致。

Internet 中平均包大小为 260Bytes，这个客户也使用这个包尺寸。

3.1.4 选择适当的 TOS 位

本用户未实施 QOS，在这个例子中不涉及到 TOS 位

3.1.5 选择适当的采样间隔

IP SLA 发送监控流量采样的频率取决于需要和监控流量本身对网络带宽的需求。采样发生的频率可以基于取得最精确的网络服务水平结论这样一个前提来考虑，但很不幸，一般这是不现实的，在一个昂贵的广域链路上，用户不会允许测试的流量占用太多带宽的。在使用低端路由器时或有大量流量通过路由器时，还得考虑产生 SLA 流量对设备本身性能的影响。在这种情况下，需要降低采样的频率或使用一个单独的路由器来做 IP SLA 作业。更细节的性能讨论，请参见本文的后续章节。

在这个例子中，我们选择了如下的采样间隔：

- UDP: 60 秒
- HTT：300 秒

3.1.6 选择适当的域值

一般情况下，服务提供商在 SLC 中都会提供一个预先定义好的性能域值，ISP 提供的 SLA 中可能包括总延时和可用性的百分比等参数。在这种情况下，选择什么域值是很清楚的。如果这些值都不明确，则需要网络管理员来决定应当选择什么样的域值。目前域值可以有响应时间和抖动计算，不适用于丢包率。

现实的域值案例:

•单向延时:

–西欧-美国西部: 90 ms

–美国西部-美国东部: 30 ms

–欧洲: 40 ms

–西欧-非洲: 150 ms

–西欧-北亚: 100 ms

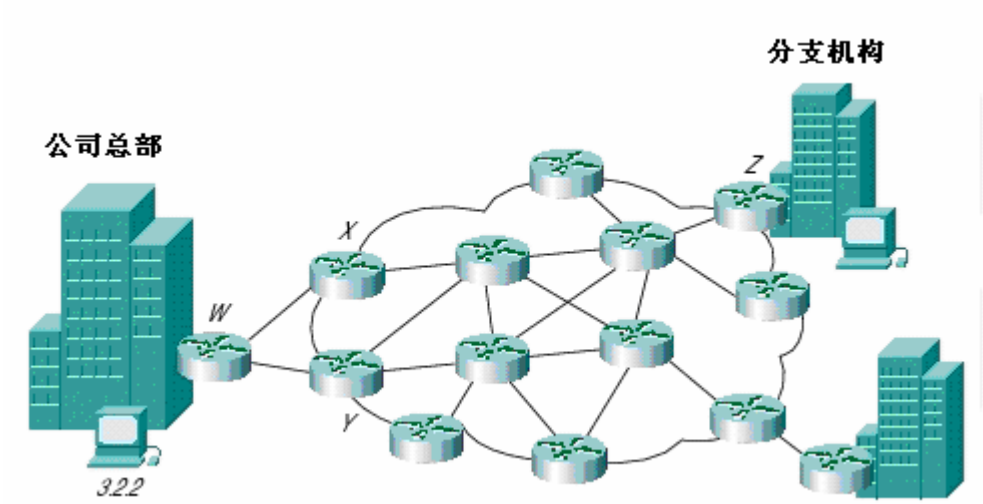
这些数据仅考虑了运营商的骨干链路部分，接入层网络增加的延时还必须增加进来

表 2：本例中配置了如下阈值

源	目的	度量	域值
A	B	往返延时 < 150 ms	>150 ms 或<100 ms
C	D	往返延时< 200 ms	>200 ms 或<150 ms
A,B,C,D	DNS	DNS 超时 10 sec	>5 s 或<3 s
A,B,C,D	Web 服务器	TCP 连接 < 500 ms	>500 ms 或＜200 ms
A,B,C,D	Web 服务器	HTTP 超时或 5 sec	>2 s 或<1 s

3.2 例 2:总部到分支机构的电话会议

图 3：例 2 网络拓朴图



3.2.1 选择适当的作业

在例 2 中，业务经理需要在总部和分支机构之间召开电话会议，客户已经通过三个不同的类在链路上布署了 QOS。在这种情况下，音频和视频在网络中极度依赖于包的延时和丢包率。为 VOIP 选择一个抖动作业，再为业务数据流量选择一个 UDP 作业。未对尽力投递类数据定义作业。

3.2.2 抖动/VOIP

VOIP 作业，通过称为抖动，度量了两个方向上面包与包之间延时的变化（源到目的/目的到源），IP SLA 以一个特定的间隔发出一系列的包，包的序号与时标及响应这些包的相关参

数被搜集起来计算中间延时的变化。该测量方式在验证 VOIP 服务中相当有用，使用抖动作业时需要在目标设备上启用 IP SLA Responder。

抖动作业与其它作业相比，提供了如下的些信息：

- 抖动：源à目的，目的à源
- 丢包情况：源à目的，目的à源
- 往返时间
- 如果 SLA 与 Responder 的时钟是同步的，可以测试出单向延时
- 最精确的作业
- 包有序列号
- 12.3(4)T 开始提供 MOS 度量与 Codec 模拟
- 12.3(7)T 开始提供单向延时、抖动、丢包及 MOS 的 Trap

3.2.3 选择适当的测试组合

- 相关路由器: W, X, Y
- 分支路由器: Z
- 作业: X 到 Z; Y 到 Z; W 到 Z

3.2.4 选择适当的荷载大小

使用 200Bytes 的包尺寸

3.2.5 选择适当的 TOS 字段

从某种意义上来说，不同类型的流量在通过网络时具有不同的优先级。如某公司坚信 Email 比 WEB 流量更为重要，它就可能为 Email 流量设置一个比 Web 流量更没的优先级。IP SLA 可以在 IP 头中配置 TOS 位。如果在 QOS 策略中是使用的 DSCP 位，DSCP 位需要转化为 TOS 位输入到 IP SLA 选项中去，因为目前的 IP SLA 特性不直接支持 DSCP 值。

表三：定义三个不同的 QOS 类

类	IP 优先级	DSCP	TOS
VoIP	101	40	160
业务数据	100	32	128
尽力投递	000	00	000

表四：定义 5 个不同的 QOS 类

类	IP 优先级	DSCP	TOS
VoIP	101	40	160
视频	100	32	128
语音控制流量	011	24	096
业务数据	001	08	032
尽力投递	000	00	000

3.2.6 选择适当的采样间隔

抖动使用 10 个间隔 20ms 的 64bytes 包作为一组，频率如下：

- X-Z: 60 sec
- Y-Z: 60 sec
- W-Z: 180 sec

3.2.7 选择适当的阈值

在这个 SLA 中没有特定的阈值，管理员必须进行独立的测试，这里选择了如下表所示的阈值。

表五：本例中所用到的域值

源	目的	度量	阈值
X—实时应用	Z	双向延时 < 100 ms 抖动 < 20ms	>100 ms 或<50 ms 抖动>20ms
X—关键应用	Z	双向延时< 500 ms	> 500 ms 或<300 ms, 超时 5 sec
W—实时应用	Z	双向延时 < 100 ms 抖动 < 20ms	>100 ms 或<50 ms 抖动>20ms
W—关键应用	Z	双向延时< 500 ms	> 500 ms 或<300 ms, 超时 5 sec
Y—实时应用	Z	双向延时 < 100 ms 抖动 < 20ms	>100 ms 或<50 ms 抖动>20ms
Y—关键应用	Z	双向延时< 500 ms	> 500 ms 或<300 ms, 超时 5 sec

G.114 标准建议单向延时(电话到电话)低于 150ms 是可接受的，我们建议延时不要过超过 20-40ms

4 IP SLA 配置与作业细节

4.1 路由器处理延时

因为还存在其它高优先级应用的原因，路由器需要花约 10 个 ms 来处理进入路由器的包。这个延时会影响到通过 Ping 技术计算出来的响应时间，因为响应包中包括到在队列中等待处理的时间，这样，测试出来的响应时间就不是真正的网络延时。IP SLA 通过使用 Responder，可以将路由器的处理延时从最终的计算结果中去掉从而保证测出来的是真正的往返延时。

4.2 Responder 与 IP SLA 控制协议

IP SLA Responder 是内置在 Cisco 路由器中的一个组件，用于响应 IP SLA 的请求包。Responder 在回显包上打上时标，从而可以计算出单向丢包、延时及抖动参数。通过使用 Responder，测试的精度也可以达到令人满意的效果。 IP SLA 的精度大大好于 ICMP ping 的精度，可以参见如下文档：

http://www.cisco.com/warp/public/cc/pd/iosw/prodlit/sanpo_wp.htm

通过使用 IP SLA Responder，可以在源和目的设备之间使用一个专用的控制协议，仅仅运行 Cisco IOS 的设备才能成为 IP SLA 的源和 IP SLA Responder 的目的设备。IP SLA Responder 可以用于 UDP 抖动作业(必须)、UDP echo 和 TCP 连接作业。当目的路由器是思科路由器时，我们建议配置 UDP echo 和 tcp 连接时均使用 Responder。而且不使用目的路由器上的 Small-server.

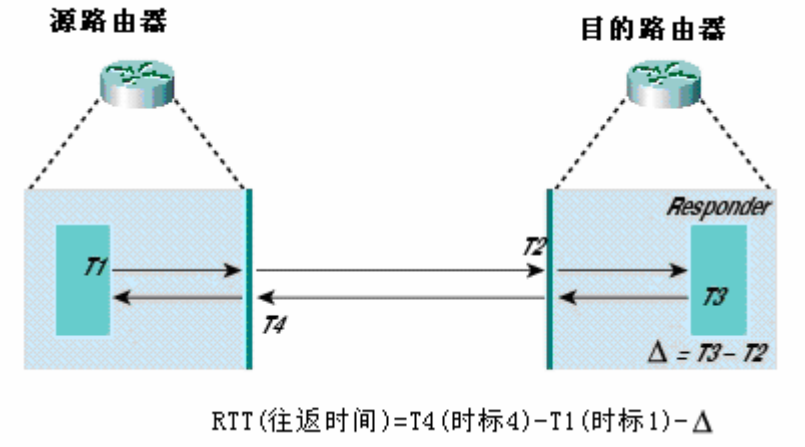
IP SLA 可以定义 Responder 所监听的端口，Responder 在一个特定的端口上监听 IP SLA 作业的控制信息。控制信息携带了如协议、端口号、持续时间等信息，一旦收到控制信息，Responder 将在特定的时间内开启特定的 UDP/TCP 端口。然后 Responder 就在该端口上接收并响应包，响应完成或超过一个预计的时间，就关闭刚才所开启的端口。为了增加 SLA 控制信息的安全性，用户可以使用 MD5 认证。

当 IP SLA 作业需要目标路由器上有 Responder 存在时会发生以下情况

- 用户初始化一个定义了目的路由器、协议及端口号的作业
- IP SLA 向目标路由器发送控制信息
- 如果启用了 MD5 验证，MD5 checksum 和控制信息一起发送
- 如果消息中的验证字段是启用的，Responder 检验验证字段
- 如果 SLA 探针没收到响应，再将发送控制信息直到超时

如果 Responder 不能处理控制信息，将返回错误，如果 Responder 处理了控制信息，它将返回一条 OK 消息到源路由器，并在消息中所定义的端口上监听。当返回码是 OK 是，源路由器开始向 Responder 发送真正的测试包。

图 4：源与 Responder 之间的度量时标



4.2.1 使用命令行启动 SLA Responder

```
(config #) rtr responder
```

4.2.2 使用 SNMP 启动 SLA Responder

```
rttMonApplResponder.0 -Integer 1
```

4.2.3 使用命令行配置控制协议的 MD5 验证

需要在源路由器和目标路由器上同时配置密钥对，命令如下：

```
(config #) key chain <name>
(config-keychain #) key <number-1>
(config-keychain-key #) key-string <authentication string>
(config-keychain-key #) exit
(config-keychain #) key <number-2>
(config-keychain-key #) key-string <authentication string>
(config-keychain-key #) exit
(config #) rtr key-chain <name>
```

4.2.4 使用 SNMP 配置 MD5 验证

当前只能创建一个认证表。

```
rttMonApplAuthStatus.<index> -Integer 4 \
rttMonApplAuthKeyChain.<index> -DisplayString "text" \
rttMonApplAuthKeyString1.<index> -DisplayString "string" \
rttMonApplAuthKeyString2.<index> -DisplayString "string"
```

一量这个表创建好，就 key-chain 和 key-strings 就不能删除，要删除只能删除个表格。

4.2.5 可伸缩的 IP SLA 布署

通常情况下，IP SLA 作业可以单独调度，每个目标的测试都需要使用 rtr schedule 命令来进行调度，顺序执行大量的 SLA 作业是 IP SLA 取得良好性能的关键之一。考虑一个有 100 个作业的源，要将这 100 个作业调度到不同时间去执行显然并不是一件很容易的事。从 IOS 12.3(8)T 开始，提供了多作业调度特性。具体的命令可以参见相关的文档。

4.3 UDP 作业

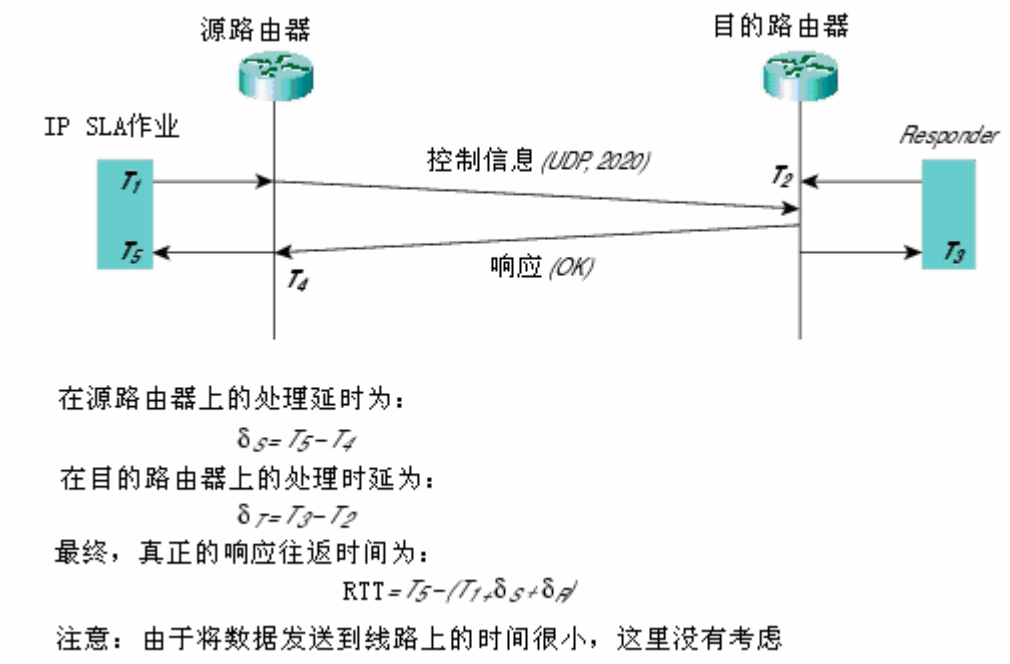
UDP 作业计算思科路由器和一个 IP 设备之间的 UDP 响应时间，响应时间通过计算发送一个数据报到目标设备和从目标设备收到响应之间的间隔来实现。如果目标设备是思科路由器，缺省情况下 UDP small-server 是关闭的，用户可以选择通过 service udp-small-server 命令启用系统 IOS 自带的小的 udp 服务器。也可以选择启动 IP SLA Responder 以提高测试精度。

UDP 作业在使用缺省的 UDP echo port(UDP port 7)，在使用 IP SLA Responder 的情况下，可以自定义端口。

4.3.1 响应时间计算

如图 5 所示，用户在计算时可以最小化处理延时，并将它从总的时延中减去。

图 5 响应时间计算



4.3.2 配置 UDP Echo Operation

```
(config)# rtr 1
```

```
(config-rtr)#type udpecho dest-ipaddr 100.100.100.2 dest-port
5000
(config)#rtr sch 1 start-time now
```

4.3.3 检查 UDP Echo 的统计信息

```
R1#show rtr op
  Entry number: 1
  Modification time: *13:26:52.947 PST Wed Jun 2 2004
  Number of operations attempted: 7
  Number of operations skipped: 0
  Current seconds left in Life: 3227
  Operational state of entry: Active
  Last time this entry was reset: Never
  Connection loss occurred: FALSE
  Timeout occurred: FALSE
  Over thresholds occurred: FALSE
  Latest RTT (milliseconds): 104
  Latest operation start time: *13:32:52.955 PST Wed Jun 2 2004
  Latest operation return code: OK
```

表六：UDP echo 统计信息说明

字段	描述
Entry number	IP SLA 作业号
Modification time	本作业的创建时间
Number of operations attempted	已发送的作业数量
Number of operations skipped	跳过的作业数量
Current seconds left in life	本作业在停止前还能运行的时间，这是一个可配置的参数
Operational state of entry	作业是否是活动的，并在测试网络
Last time this entry was reset	作业被复位的最新时间，复位可以清空所有的的统计值
Connection loss occurred	是否发生过连接丢失
Timeout occurred	是否发生过超时
Over thresholds occurred	设置了阈值，并出现了超时现象
Latest RTT (milliseconds)	最后一次测试的往返时间
Latest operation start time	最后一次测试开始时间
Latest operation return code	作业状态

4.4 UDP 抖动作业

在网络中有实时流量存在的情况下，度量网络性能就不仅仅只考虑可用性了，更多的会关注网络中的延时等参数，实时应用与延时是紧密相关的。对于语音数据，丢包是比较容易处理的，但频繁的丢包会引起通话质量的下降。UDP 抖动作业能在一个作业中提供丢包情况、抖动和延时参数，同时也可以很好的测试单向的参数。

抖动作业被设计为通过产生活动的 UDP 流量来测试网络中的延时、延时变化（抖动）丢包率等参数。它每次从源路由器发送 N 个大小为 S byte 的包到目的路由器，包之间的间隔为 T ms，所有的这些参数都是用户可以配置的。

基于抖动作业发出的包/接收到的包中所携带的时标及序号信息，就可以测量如每个方向的延时变化(抖动)，每个方向的丢包率，平均的往返时间，单向延时[需要 12.2(2)T 或更新的 IOS]等参数值。

4.4.1 抖动计算

源以 10ms 的间隔连续的向目的发出测试包，如果网络运行处于理想状态，目的应该以 10ms 的间隔收到这一系列的包。队列、使用备用路由等因素所导致的延时会让包到达目的路由器的间隔大于或小于 10ms。正的抖动意味着两个包到达的间隔大于 10ms，如间隔 12ms，这时正抖动就是 2ms，同理负抖动是指到达间隔小于原始间隔。在语音网络中不希望有太大的抖动值，在一个对延时敏感的网络中，最理想的情况是抖动为 0。

4.4.2 单向延时计算

理论上，包从主机 A 到主机 B 所花的时间在两个方向上应该完全一致，但在实际环境中，两个方向上的延时可能有很大的区别，有可能一个方向远远大于别一个方向。考虑一条繁忙的高速公路，完全有可能在某一个方向上发生堵车的现象，在网络中也存在同样的现象。在源和目的之间也可能存在非对称的路径。单向延时计算为用户提供了网络中更详细的性能。用户可以更方便的理解网络中的瓶颈究竟在哪里。

UPD 抖动作业提供了单向延时的测试能力。然而，单向延时的测试需要源和目的路由器上的时钟要相当同步才行。这时需要使用基于 GPS 的 NTP 服务器，当源与目的路由器的时钟不同步时，IP SLA 会忽略单向延时的计算，并填上 0。单向抖动和丢包率不需要时钟同步。

4.4.3 通过命令行配置抖动作业

```
(Config)#rtr 200
(config-rtr)#type jitter dest-ip 172.24.132.100 dest-port 99
num-packets 20 interval 20
```

4.4.4 针对抖动作业使用 show 命令

```
R1#show rtr collection-statistics
    Entry Number: 1
    Target Address: 172.24.132.100, Port Number: 31337
    Start Time: *14:14:14.000 EST Thu Apr 6 2000
    RTT Values:
    NumOfRTT: 2800 RTTSum: 4792 RTTSum2: 8830
```

```
Packet Loss Values:
PacketLossSD: 0 PacketLossDS: 0
PacketOutOfSequence: 0 PacketMIA: 0 PacketLateArrival: 0
InternalError: 0 Busies: 0
Jitter Values:
MinOfPositivesSD: 1 MaxOfPositivesSD: 1
NumOfPositivesSD: 249 SumOfPositivesSD: 249 Sum2PositivesSD: 249
MinOfNegativesSD: 1 MaxOfNegativesSD: 2
NumOfNegativesSD: 238 SumOfNegativesSD: 239 Sum2NegativesSD: 241
MinOfPositivesDS: 1 MaxOfPositivesDS: 1
NumOfPositivesDS: 97 SumOfPositivesDS: 97 Sum2PositivesDS: 97
MinOfNegativesDS: 1 MaxOfNegativesDS: 1
NumOfNegativesDS: 92 SumOfNegativesDS: 92 Sum2NegativesDS: 92
Interarrival jitterout: 0 Interarrival jitterin: 0
One Way Values:
NumOfOW: 0
OWMinSD: 0 OWMaxSD: 0 OWSumSD: 0 OWSum2SD: 0
OWMinDS: 0 OWMaxDS: 0 OWSumDS: 0 OWSum2DS: 0
```

表 7：show rtr collection-statistics 的各域的说明

字段	说明
NumOfRTT	成功的往返次数
RTTSum	全程时间总和
RTTSum2	全程时间平方和
PacketLossSD	源到目的丢失的包
PacketLossDS	目的到源丢失的包
PacketOutOfSequence	返回包序列号不正确的数量
PacketMIA	出现丢包，但方向不能确定，这通讯是一个测试流的最后一个包丢了。
PacketLateArrival	在超时之后到达的包的数量
InternalError	由于其它内部错误导致作业不能正常运行的次数
Busies	由于上次运行未完成而导致本次不能运行的次数＝
MinOfPositivesSD MaxOfPositivesSD	源到目的的最小和最大正抖动（单位为 MS）
NumOfPositivesSD	源到目的正抖动数量
SumOfPositivesSD	源到目的总的抖动值
Sum2PositivesSD	源到目的正抖动的平方和
MinOfNegativesSD MaxOfNegativesSD	目的到源的最小和最大正抖动（单位为 MS）
NumOfNegativesSD	目的到源正抖动数量
SumOfNegativesSD	目的到源总的抖动值
Sum2NegativesSD	目的到源正抖动的平方和

4.5 应用于 VoIP 的 UDP 抖动作业

这一类 UDP 抖动作业是在当前的 UDP 抖动作业基础上的一个增强与扩展，经过增强与扩展之后，经命令行或 MIB 配置，可以通过模拟 codec 来实现对语音质量的测试并直接得出语音质量评分，目前 12.3(4)T 可以支持如下几种 codec:

- 711 A Law
- G.711 u Law
- G.729A

命令如下：

```
(config)# Rtr 1
(config-rtr)# type jitter dest-ipaddr <ipaddress> dest-port
<portno> codec <codectype> codec-interval <value> codec-size
<value> codec-numpacket <value>
```

新的参数的选项如下表所示：

Codectype	codec-size	codec-interval	codec-numpackets
G711ulaw	172	20 ms	1000
G711alaw	172	20 ms	1000
G729a	32	20 ms	1000

VoIP 作业使用 ICPIF 语音质量评价系统事测试 MOS 值，可以通过命令行和 MIB 存取活动的测试结果中的 MOS 值。这个测试可以从应用的角度看出来网络性能参数对 VoIP 的支持程度。虽然 ICPIF/MOS 参数在评价语音质量时是非常有用的，但它并没有包括能够影响到语音质量的所有参数。更详细的信息可以参见如下的文档：
http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_4/gtsaamos.htm#1043332

IOS 12.3(7)T 提供了基于单向抖动、丢包情况、延时及 MOS/ICPIF 语音得分参数设置反应阈值或发出 SNMP trap 的能力，详细情况可以参见如下链接：
http://www.cisco.com/en/US/products/sw/iosswrel/ps5207/products_feature_guide09186a008020a3c9.html

以下命令行在 12.3(7)T 下测试通过：

```
logging on
 rtr 10
  type jitter dest-ipaddr 209.165.200.225 dest-port 16384 codec
g711alaw advantage-
factor 2
  owner admin
  tag jitter-with-voice-scores
 rtr schedule 10 start-time now
 rtr reaction-configuration 10 react mos threshold-type immediate\
threshold-value 490 250 action-type trapOnly
 rtr logging traps
 snmp-server host 10.10.10.10 version 2c public
```

snmp-server enable traps syslog

12.3(7)T 中的 trap 的实质是 syslog 到 traps 的转换，因此需要在命令行配置 logging，在以后的版本中将提供纯粹的 SNMP trap.

下面是相关 Show 的结果

Router# show rtr operational-state 10

```
Entry number: 10
Modification time: 12:57:45.690 UTC Sun Oct 26 2003
Number of operations attempted: 1
Number of operations skipped: 0
Current seconds left in Life: Forever
Operational state of entry: Active
Last time this entry was reset: Never
Connection loss occurred: FALSE
Timeout occurred: FALSE
Over thresholds occurred: FALSE
Latest RTT (milliseconds): 19
Latest operation start time: 12:57:45.723 Sun Oct 26 2003
Latest operation return code: OK
!
Voice Scores:
ICPIF: 20 MOS Score: 3
!
RTT Values:
NumOfRTT: 10 RTTAvg: 19 RTTMin: 19 RTTMax: 20
RTTSum: 191 RTTSum2: 3649
Packet Loss Values:
PacketLossSD: 0 PacketLossDS: 0
PacketOutOfSequence: 0 PacketMIA: 0 PacketLateArrival: 0
InternalError: 0 Busies: 0
Jitter Values:
NumOfJitterSamples: 9
MinOfPositivesSD: 0 MaxOfPositivesSD: 0
NumOfPositivesSD: 0 SumOfPositivesSD: 0 Sum2PositivesSD: 0
MinOfNegativesSD: 0 MaxOfNegativesSD: 0
NumOfNegativesSD: 0 SumOfNegativesSD: 0 Sum2NegativesSD: 0
MinOfPositivesDS: 1 MaxOfPositivesDS: 1
NumOfPositivesDS: 1 SumOfPositivesDS: 1 Sum2PositivesDS: 1
MinOfNegativesDS: 1 MaxOfNegativesDS: 1
NumOfNegativesDS: 1 SumOfNegativesDS: 1 Sum2NegativesDS: 1
Interarrival jitterout: 0 Interarrival jitterin: 0
One Way Values:
NumOfOW: 0
OWMinSD: 0 OWMaxSD: 0 OWSumSD: 0 OWSum2SD: 0
OWMinDS: 0 OWMaxDS: 0 OWSumDS: 0 OWSum2DS: 0
```

Router# show rtr collection-statistics 10

```
Entry number: 10
Start Time Index: 12:57:45.931 UTC Sun Oct 26 2003
Number of successful operations: 60
Number of operations over threshold: 0
Number of failed operations due to a Disconnect: 0
Number of failed operations due to a Timeout: 0
Number of failed operations due to a Busy: 0
Number of failed operations due to a No Connection: 0
Number of failed operations due to an Internal Error: 0
Number of failed operations due to a Sequence Error: 0
Number of failed operations due to a Verify Error: 0
Voice Scores:
MinOfICPIF: 2 MaxOfICPIF: 20 MinOfMos: 3 MaxOfMos: 5
RTT Values:
NumOfRTT: 600 RTTAvg: 20 RTTMin: 19 RTTMax: 22
RTTSum: 12100 RTTSum2: 244292
Packet Loss Values:
PacketLossSD: 0 PacketLossDS: 0
```

```
PacketOutOfSequence: 0 PacketMIA: 0 PacketLateArrival: 0
InternalError: 0 Busies: 0
Jitter Values:
NumOfJitterSamples: 540
MinOfPositivesSD: 1 MaxOfPositivesSD: 1
NumOfPositivesSD: 26 SumOfPositivesSD: 26 Sum2PositivesSD: 26
MinOfNegativesSD: 1 MaxOfNegativesSD: 1
NumOfNegativesSD: 19 SumOfNegativesSD: 19 Sum2NegativesSD: 19
MinOfPositivesDS: 1 MaxOfPositivesDS: 1
NumOfPositivesDS: 43 SumOfPositivesDS: 43 Sum2PositivesDS: 43
MinOfNegativesDS: 1 MaxOfNegativesDS: 2
NumOfNegativesDS: 43 SumOfNegativesDS: 44 Sum2NegativesDS: 46
Interarrival jitterout: 0 Interarrival jitterin: 0
One Way Values:
NumOfOW: 0
OWMinSD: 0 OWMaxSD: 0 OWSumSD: 0 OWSum2SD: 0
OWMinDS: 0 OWMaxDS: 0 OWSumDS: 0 OWSum2DS: 0.
```

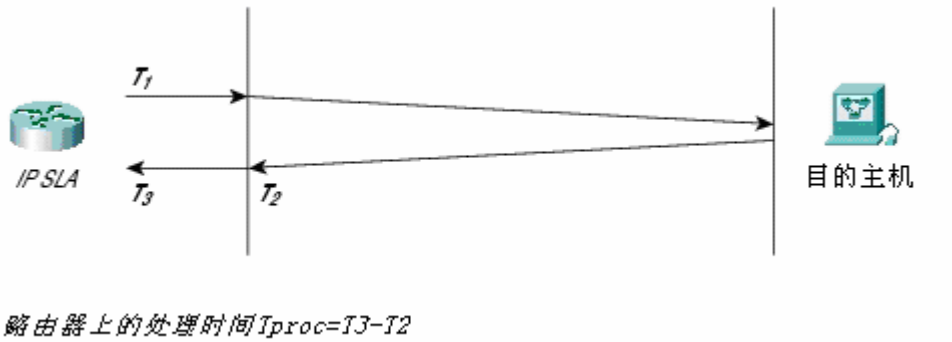
4.6 ICMP Echo 作业

ICMP Echo 作业可以测试思科路由器和 IP 设备之间的端到端的响应时间，响应时间是通过计算 ICMP echo 请求与响应消息的时间差也教育处的。IP SLA 通过在 IP 包中设置 DSCP 位，也允许用户测量 QOS 的实施情况。

4.6.1响应时间计算

如以前讨论的一样，在计算时，处理延时被最小化并在最终结果中减去。

图 6：ICMP echo 作业的响应时间计算



4.6.2 ICMP 荷载

通过设置”request size”用户可以配置 ICMP ECHO 作业的有效荷载尺寸，路由器将在定义的数值之上加上 36 Bytes，如定义的 request-size 是 28Bytes，则实际的 ICMP 包的大小是 64Bytes。

4.6.3 配置 ICMP echo 作业

```
(config)#rtr 2
```

```
(config-rtr)#type echo protocol ipicmpecho 100.100.100.2
(config-rtr-echo)#request-data-size 400
(config-rtr-echo)#tos 160
(config)#rtr sch 2 start now
```

4.6.4 ICMP echo 作业的 show 结果

```
R1#show rtr operation 2
    Entry number: 2
    Modification time: *13:51:09.195 PST Wed Jun 2 2004
    Number of operations attempted: 1
    Number of operations skipped: 0
    Current seconds left in Life: 3545
    Operational state of entry: Active
    Last time this entry was reset: Never
    Connection loss occurred: FALSE
    Timeout occurred: FALSE
    Over thresholds occurred: FALSE
    Latest RTT (milliseconds): 0
    Latest operation start time: *13:51:09.203 PST Wed Jun 2 2004
    Latest operation return code: OK
    RTT Values:
    RTTAvg: 0 RTTMin: 0 RTTMax: 0
    NumOfRTT: 1 RTTSum: 0 RTTSum2: 0
```

表 8：ICMP echo 作业的 show rtr operation 域描述：

字段	描述
Latest RTT	最后一次的往返时间(ms)
RTTAvg	平均往返时间(ms)
RTTMin	最小往返时间(ms)
RTTMax	最大往返时间(ms)
NumofRTT	已测试的往返个数
RTTSum	往返时间之和
RTTSum2	往返时间的平方和

4.7 ICMP PATH echo 作业

ICMP path echo 作业可以测试出思科路由器到任 IP 设备之间的每一跳的响应时间，它使用 *traceroute* 命令来发现到目的的路径，然后测试源到路径上每一个中间跳的响应时间。如果到目的地有多条等价路径，通过在中间路由器上设置 LSR 参数，SLA 可以选择一条特定的路径进行测试。基本的配置命令如下：

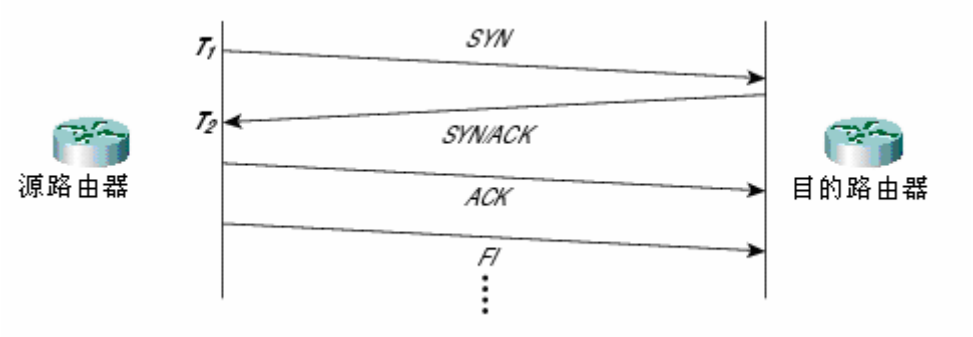
```
Router#
 rtr 3
 type pathEcho protocol ipIcmpEcho <ip_address>
 frequency 10
 lives-of-history-kept 5
```

```
buckets-of-history-kept 3
filter-for-history all
rtr schedule 3 life 25 start-time now
```

4.8 TCP 连接作业

TCP 连接作业可以测量出从思科路由器到任意 IP 设备之间进行 TCP 连接所需要的时间，如目标设备是思科路由器，用户可以在目标路由器上启动 IP SLA Responder。如果目标设备不是思科路由器，用户必须指定一个周知端口，如 21、23、80 等。

图 7：TCP 连接响应时间



4.8.1 使用命令行配置 TCP 连接作业

创建一个 tcp 连接作业，不需要 IP SLA Responder，目的端口为 80，命令如下，*control disable* 意味着不需要 Responder

```
(config)# rtr 1
(config-rtr)# type tcpConnect dest-ipaddr 5.0.0.2 dest-port 80
control disable
(config)# rtr schedule 1 start-time now
```

创建一个需要 Responder 的 tcp 连接作业，命令如下：

```
(config)# rtr 1
(config-rtr)# type tcpConnect dest-ipaddr 5.0.0.2 dest-port 8008
(config-rtr)# tos 4
(config)# rtr schedule 1 start-time now
```

4.8.2 TCP 连接作业的 show 输出

```
Router#sh rtr op 1
    Current Operational State
    Entry Number: 1
    Modification Time: 14:17:10.000 CET Thu Aug 22 2002
    Diagnostics Text:
    Last Time this Entry was Reset: Never
    Number of Octets in use by this Entry: 1490
    Connection Loss Occurred: FALSE
    Timeout Occurred: FALSE
    Over Thresholds Occurred: FALSE
    Number of Operations Attempted: 17
    Current Seconds Left in Life: 2603
```

```
Operational State of Entry: active
Latest Completion Time (milliseconds): 6
Latest Operation Start Time: 14:33:10.000 CET Thu Aug 22 2002
Latest Operation Return Code: ok
Latest 10.52.132.68
```

4.9 DNS 作业

DNS 作业用于测试 DNS 的响应时间，IP SLA DNS 作业既可以测试从主机名到 IP 地址的查询时间，也可以测试从不 IP 地址到主机名的查询。配置命令如下：

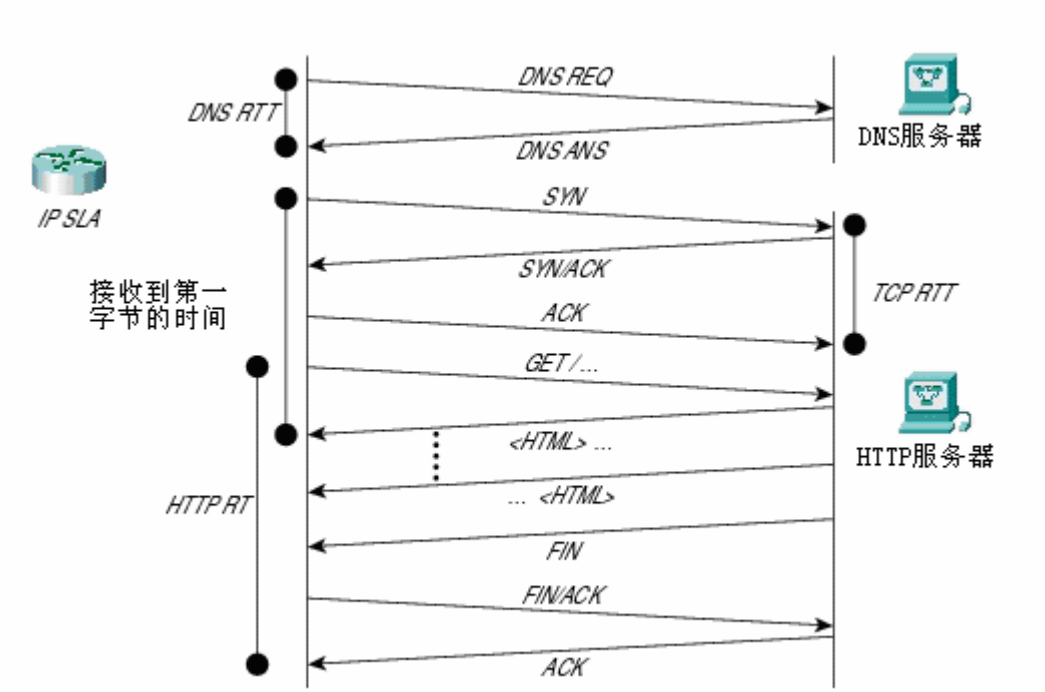
```
 (config)#rtr 1
(config-rtr)# type dns target-addr foo.foo.com name-server
10.52.128.30
(config)#rtr schedule 1 start-time now
```

4.10 HTTP 作业

HTTP 作业可以度量从一个 HTTP 服务器访问数据所需要的时间，HTTP 服务器响应时间分为如下几个部分：域名查询、TCP 连接、发现 HTTP 请求到收到 HTTP 响应这样三个时间。IP SLA 支持三种类型的 HTTP 作业，分别描述如下：

- GET 请求：基于定义的 URL 进行格式化
- RAW 模式：可以很灵活的的 RAW 数据的方式定义 HTTP 请求。可以定义认证信息等参数。
- 针对特定的 URL，取得第一个响应字节所需要的时间。

图 8：HTTP 响应时间定义



4.10.1 使用命令行定义 HTTP get 作业

```
(config)#rtr 1
(config-rtr)#type http operation get url http://foo.foo.com/foo/
(config)#rtr schedule 1 start-time now
```

Show 的输出如下：

```
Router#sh rtr op 1
     Entry number: 1
     Modification time: *22:01:31.895 MET Sun Apr 11 1993
     Number of operations attempted: 1
     Number of operations skipped: 0
     Current seconds left in Life: 3592
     Operational state of entry: Active
     Last time this entry was reset: Never
     Connection loss occurred: FALSE
     Timeout occurred: FALSE
     Over thresholds occurred: FALSE
     Latest RTT (milliseconds): 193
     Latest operation start time: *22:01:31.902 MET Sun Apr 11 1993
     Latest operation return code: OK
     Latest DNS RTT: 4
     Latest TCP Connection RTT: 8
     Latest HTTP Transaction RTT: 181
     Latest HTTP Status: 200
     Latest HTTP Message Size: 2842
     Latest HTTP Entity-Body size: 2677
```

相关字段意义描述：

字段	描述
Latest RTT	DNS、TCP 连接、HTTP 处理的响应时间之和
Latest DNS RTT	DNS 查询所响应时间
Latest TCP Connection RTT	TCP SYN 到 ACK 的响应时间
Latest HTTP Transaction RTT: 181	取得文件第一字节所花的 HTTP 处理时间
Latest HTTP Status: 200	HTTP 完成所需要的时间

4.10.2 使用命令行定义 HTTP 的 RAW 作业

```
 (config)# rtr 6
(config-rtr)# type http operation raw url http://6.0.0.2
(config-rtr)# http-raw-request
(config-rtr-http)# GET /index.html HTTP/1.0\r\n
(config-rtr-http)# \r\n
(config-rtr-http)# exit
```

4.10.3 使用命令行定义一个通过代理服务器的 RAW 作业

本例中，1.1.1.1 是代理服务器，2.2.2.2 是实际需要访问的服务器。

```
(config)# rtr 6
(config-rtr)# type http operation raw url http://1.1.1.1
(config-rtr)# http-raw-request
(config-rtr-http)# GET http://2.2.2.2/index.html HTTP/1.0 \r\n
```

```
(config-rtr-http)# \r\n
(config-rtr-http)# exit
(config)# rtr schedule 6 start-time now
```

4.10.4 使用命令行配置一个需要认证的 HTTP RAW 作业

```
(config)#rtr 1
 (config-rtr)#type http operation raw url http://foo.foo.com
(config-rtr)#http-raw-request
(config-rtr-http)#GET /lab/index.html HTTP/1.0\r\n
(config-rtr-http)#Authorization: Basic btNpdGT4biNvoZe=\r\n
(config-rtr-http)#\r\n
(config-rtr-http)#exit
(config)#rtr schedule 1 start-time now
```

4.11 DHCP 作业

DHCP 作业可以测试出从了现一个 DHCP 服务器到从该 DHCP 服务器获得一个地址所需要的时间，在完成作业后，IP SLA 将释放租用到的地址。配置 DHCP 作业时，有两种模式可以选择，缺省情况下从路由器的所有接口发出 DHCP 的 discovery 包，当配置了 ip dhcp server 命令时，discovery 包只发到所定义的 DHCP 服务器。

4.12 DLSW 作业

DLSW+是思科在 RFC1795 基础上的一个增强，该协议使用 TCP 在 IP 骨干网上传送 SNA 数据。路由器这这中间充当 DLSw peer 的角色。

IP SLA 中的 DLSw+作业可以测试在一对 DLSw peer 之间的网络响应时间，DLSw peer 一般使用 2065 的 TCP 端口。DLSw+作业只需要对端支持 RFC1795，不需要对端是思科的路由器。

4.13 FTP 作业(12.1(1)T 以后的 IOS 支持)

FTP 往往会携带大带的流量，这个作业可以测试出从主机到路由器的 FTP 吞吐量和响应时间，可以很好的刻画出网络的能力来。

4.13.1 使用命令行定义一个 FTP 作业

```
(config)#rtr 1
(config-rtr)#type ftp operation get \url
ftp://user:pwd@ftp.foo.com/test.cap
(config)#rtr schedule 1 start-time now
```

4.13.2 FTP 作业的 show 结果

```
RT#sh rtr op 1
     Entry number: 1
     Modification time: *04:02:36.295 MET Mon Apr 12 1993
     Number of operations attempted: 6
     Number of operations skipped: 0
     Current seconds left in Life: 3287
     Operational state of entry: Active
     Last time this entry was reset: Never
     Connection loss occurred: FALSE
     Timeout occurred: FALSE
     Over thresholds occurred: FALSE
     Latest RTT (milliseconds): 4193
     Latest operation start time: *04:07:36.299 MET Mon Apr 12 1993
     Latest operation return code: OK
     Bytes read: 2048000
```

相关字段定义：

字段	描述
Latest RTT	取得文件所花的响应时间
Bytes read	读取文件大小

通过表中的两个参数可以计算出下载的带宽来：

2048000 bytes / 4.193 s = 488.4 KB/s

4.14 路径抖动作业(自 12.2T 开始支持)

IOS 12.2T 开始提供的路径抖动作业可以测试出网络中每一跳的抖动、丢包率和延时参数，这个作业先通过 traceroute 发现从源到目的的路径，然后 使用 ICMP echo 来测试出每一跳响应时间、丢包率及大约的抖动参数(基于 RFC 1889)。由于 ICMP 只包括一个往返时间，因此这个作业的测试结果的精确度不高。该作业不能使用 RTTMON MIB 来进行配置与读取数据，因此只能通过命令行来进行配置和查看结果。

4.14.1 路径抖动的配置命令

```
(config)#rtr 3
(config-rtr)#type icmpPathJitter dest-ipaddr <dest-ip>
source-ipaddr <src_ip> num-packets <n> interval <t>
```

在未定义 num-packets 及 interval 值时，缺省为情况下包数量为 10 个，间隔为 20ms

4.14.2 路径抖动的 show 结果

```
---- Path Jitter Statistics ----
Source IP    - 172.17.246.5
Destination IP  - 172.17.246.20
Number of Echos - 10
Interval between Echos  - 20 ms
```

```
Target Only - Disabled (default)

Hop IP 172.17.246.2:
 RTT:1   PacketLoss:0    Jitter:0
 MinRTT:1    MaxRTT:2    SumRTT:19   Sum2RTT:37
 MinPosJitter:1 MaxPosJitter:1  SumPos:1    Sum2Pos:1
 MinNegJitter:0 MaxNegJitter:0  SumNeg:0    Sum2Neg:0
 OutOfSequence:0    DiscardedSamples:0
Hop IP 172.17.246.20:
 RTT:1   PacketLoss:0    Jitter:0
 MinRTT:1    MaxRTT:3    SumRTT:14   Sum2RTT:24
 MinPosJitter:2 MaxPosJitter:2  SumPos:2    Sum2Pos:4
 MinNegJitter:1 MaxNegJitter:1  SumNeg:2    Sum2Neg:2
 OutOfSequence:0    DiscardedSamples:0
```

4.15 使用命令查看作业状态

使用如下命令查看 SLA 是否正确配置：

show rtr application
show rtr configuration

使用如下命令查看作业运行的结果

show rtr operational-state
show rtr distributions-statistics
show rtr collection-statistics
show rtr totals-statistics

5 IP SLA 的硬件及软件需求

5.1 IP SLA 与 IOS 版本之间的关系

各版本的 IOS 所支持的 IP SLA 作业特性

特性/版本	11.2	12.0 (3)T	12.0 (5)T 12.0 (8)S	12.1E	12.1 (1)T 12.2	12.2 (2)T	12.2 (11)T (Infra2)	12.2 (14)S	12.2 (25)S
ICMP Echo	X	X	X	X	X	X	X	X	X
ICMP Echo Path	X	X	X	X	X	X	X	X	X
SSCP(SNA)	X	X	X	X	X	X	þ	X	
UDP Echo		X	X	X	X	X	X	X	X
TCP Connect		X	X	X	X	X	X	X	X
UDP Jitter			X	X	X	X	X	X	X
HTTP			X	X	X	X	X	X	X
DNS			X	X	X	X	X	X	X
DHCP			X	X	X	X	X	X	X
DLSw+			X	X	X	X	X	X	X
SNMP Support			X	X	X	X	X	X	X

UDP Jitter One Way Latency							X	X	X
FTP Get					X	X	X	X	X
MPLS/VPN Aware						X	X		X
Frame-Relay (CLI)						X	X		X
ICMP Path Jitter						X	X		X
APM						X	X		X

5.2 IP SLA 的硬件支持情况

IP SLA 在除 IP Lite 外的所有 IOS 软件特性集中支持。IOS 12.3 版本使用了 package 来划分特性集，除 IP BASE 以外，其它的 package 均支持 IP SLA。将在 12.4T 的 IP BASE 中集成进 SLA Responder 和 ICMP 作业的支持。除 4500 系列交换机外，所有支持 IOS 软件的思科硬件均支持 IP SLA。Linksys 的设备将在未来提供 Responder 的支持。

5.3 IP SLA 的软件架构变迁

从 12.2(15)T2、12.3(3)、12.2S(25)、12.0(34)S 开始，整个 IP SLA 的软件架构将进行重写，新的架构称为 Infrastructure II，新的架构增强了性能，降低了 IP SLA 的内存占用。其它的增加还包括提高了精确度，提供基于 NTP 的时标等。Infrastructure II 还包括如下增强：

- 自 12.3(3)开始，IP SLA 作业的数量仅受 CPU 能力和内存的限制。在以前的版本中，12.2(11)T 以前只能配置 500 个作业，12.2(11)T 以后也只能配置 2000 个作业。
- SLA 对内存的占用降低了 50%
- 新的架构所有的作业使用同一进程，在以前，每个作业一个进程。

6 IP SLA RTTMon MIB

截止 2004 年 9 月，Cisco-RTTMON-MIB.my，只有 FR、Path Jitter、ATM 三种作业不能支持。

6.1 参考文档

不同硬件及 IOS 版本对 MIB 的支持程度：
http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

思科 MIB 查询工具

http://tools.cisco.com/ITDIT/MIBS/servlet/index

SNMP OID 查询(需要 CCO 账号)：

http://www.cisco.com/cgi-bin/Support/Mibbrowser/unity.pl

6.2 使用 RTTMON MIB 创建作业

通过 SNMP 创建 IP SLA 作业可以使用两个命令：createAndGo 和 createAndWait。每个令在执行前均需要定义如下一些变量：

```
set rttMonCtrlAdminStatus
set rttMonCtrlAdminRttType
set rttMonEchoAdminProtocol
```

对于每个作业，在它被提交前，还有一些作业相关的参数需要设置，如下所示：

作业	参数
echo, pathEcho 及 dlsw	rttMonEchoAdminTargetAddress
udpEcho、tcpConnect 和 jitter	rttMonEchoAdminTargetAddress rttMonEchoAdminTargetPort
HTTP	rttMonEchoAdminURL
DNS	rttMonEchoAdminTargetAddressString rttMonEchoAdminNameServer

6.2.1 ECHO 作业示例

```
rttMonCtrlAdminStatus.<index> -Integer 4 \
rttMonCtrlAdminRttType.<index> -Integer 1 \
rttMonEchoAdminProtocol.<index> -Integer 2 \
rttMonEchoAdminTargetAddress.<index> -OctetString "04 00 00 01" \
rttMonEchoAdminSourceAddress.<index> -OctetString "01 00 00 01" \
rttMonEchoPathAdminHopAddress.<index>.1 -OctetString "02 00 00 01" \
rttMonEchoPathAdminHopAddress.<index>.2 -OctetString "03 00 00 01" \
rttMonScheduleAdminRttStartTime.<index> -TimeTicks 1 \
rttMonScheduleAdminRttLife.<index> -Integer 2147483647
```

6.2.2 Path Echo 作业

```
rttMonCtrlAdminStatus.<index> -Integer 4 \
rttMonCtrlAdminRttType.<index> -Integer 2 \
rttMonEchoAdminProtocol.<index> -Integer 2 \
rttMonEchoAdminTargetAddress.<index> -OctetString "05 00 00 02" \
rttMonScheduleAdminRttStartTime.<index> -TimeTicks 1 \
rttMonScheduleAdminConceptRowAgeout.<index> -Integer 0
```

6.2.3 UDP 作业

```
rttMonCtrlAdminStatus.<index> -Integer 4 \
rttMonCtrlAdminRttType.<index> -Integer 5 \
rttMonEchoAdminProtocol.<index> -Integer 3 \
rttMonEchoAdminTargetAddress.<index> -OctetString "05 00 00 02" \
rttMonEchoAdminTargetPort.<index> -Integer 4444 \
rttMonEchoAdminTOS.<index> -Integer 5 \
rttMonScheduleAdminRttStartTime.<index> -TimeTicks 1
```

6.2.4 TCP 作业

```
rttMonCtrlAdminStatus.<index> -Integer 4 \
rttMonCtrlAdminRttType.<index> -Integer 6 \
rttMonEchoAdminProtocol.<index> -Integer 24 \
rttMonEchoAdminTargetAddress.<index> -OctetString "05 00 00 02" \
rttMonEchoAdminTargetPort.<index> -Integer 80 \
rttMonEchoAdminControlEnable.<index> -Integer 2 \
rttMonScheduleAdminRttStartTime.<index> -TimeTicks 1
```

6.2.5 抖动作业

```
rttMonCtrlAdminStatus.<index> -Integer 4 \
rttMonCtrlAdminRttType.<index> -Integer 9 \
rttMonEchoAdminProtocol.<index> -Integer 27 \
rttMonEchoAdminTargetAddress.<index> -OctetString "05 00 00 02" \
rttMonEchoAdminTargetPort.<index> -Integer 8000 \
rttMonEchoAdminInterval.<index> -Integer 20 \
rttMonEchoAdminNumPackets.<index> -Integer 100 \
rttMonScheduleAdminRttStartTime.<index> -TimeTicks 1
```

6.2.6 简单的 HTTP GET 作业

```
rttMonCtrlAdminStatus.<index> -Integer 4 \
rttMonCtrlAdminRttType.<index> -Integer 7 \
rttMonEchoAdminProtocol.<index> -Integer 25 \
rttMonEchoAdminOperation.<index> -Integer 1 \
rttMonEchoAdminURL.<index> -DisplayString
"http://www.foo.com:80/index.html" \
rttMonEchoAdminHTTPVersion.<index> -DisplayString "1.0" \
rttMonEchoAdminCache.<index> -Integer 2 \
rttMonScheduleAdminRttStartTime.<index> -TimeTicks 1
```

6.2.7 HTTP RAW 作业

```
rttMonCtrlAdminStatus.<index> -Integer 4 \
rttMonCtrlAdminRttType.<index> -Integer 7 \
rttMonEchoAdminProtocol.<index> -Integer 25 \
rttMonEchoAdminOperation.<index> -Integer 2 \
rttMonEchoAdminURL.<index> -DisplayString "http://www.foo.com" \
rttMonEchoAdminString1.<index> -DisplayString "GET /index.html
HTTP/1.0\r\n\r\n" \
rttMonEchoAdminNameServer.<index> -OctetString "01 05 07 09" \
rttMonScheduleAdminRttStartTime.<index> -TimeTicks 1
```

6.2.8 HTTP RAW 作业，使用代理服务器，带认证

```
rttMonCtrlAdminRttType.1 = 7
rttMonEchoAdminProtocol.1 = 25
rttMonEchoAdminOperation.1 = 2
rttMonScheduleAdminRttStartTime.1 = 1
rttMonEchoAdminURL.1 = http://kick
rttMonEchoAdminString1.1 = GET / HTTP/1.0\r\n
rttMonEchoAdminString2.1 = Authorization: Basic aXBtOmNpc2Nv\r\n
rttMonCtrlAdminStatus.1 = 4
```

使用 snmpset 命令进的完整命令格式为：

```
snmpset -v2c -c public 10.52.132.69 \
 .1.3.6.1.4.1.9.9.42.1.2.1.1.4.1 i 7 \
 .1.3.6.1.4.1.9.9.42.1.2.2.1.1.1 i 25 \
 .1.3.6.1.4.1.9.9.42.1.2.2.1.13.1 i 2 \
```

```
.1.3.6.1.4.1.9.9.42.1.2.5.1.2.1 t 1 \
.1.3.6.1.4.1.9.9.42.1.2.2.1.15.1 s "http://kick" \
.1.3.6.1.4.1.9.9.42.1.2.2.1.20.1 s "GET / HTTP/1.0\r\n" \
.1.3.6.1.4.1.9.9.42.1.2.2.1.21.1 s "Authorization: Basic \
aXBtOmNpc2Nv\r\n" \
.1.3.6.1.4.1.9.9.42.1.2.1.1.9.1 i 4
```

6.2.9 DNS 作业

```
rttMonCtrlAdminStatus.<index> -Integer 4 \
rttMonCtrlAdminRttType.<index> -Integer 8 \
rttMonEchoAdminProtocol.<index> -Integer 26 \
rttMonEchoAdminTargetAddressString.<index> -DisplayString
"www.cisco.com" \
rttMonEchoAdminNameServer.<index> -OctetString "11 05 07 09" \
rttMonScheduleAdminRttStartTime.<index> -TimeTicks 1
```

6.2.10 DLSW 作业

```
rttMonCtrlAdminStatus.<index> -Integer 4 \
rttMonCtrlAdminRttType.<index> -Integer 10 \
rttMonEchoAdminProtocol.<index> -Integer 28 \
rttMonEchoAdminTargetAddress.<index> -OctetString "05 00 00 02" \
rttMonScheduleAdminRttStartTime.<index> -TimeTicks 1
```

6.2.11 DHCP 作业

```
rttMonCtrlAdminStatus.<index> -Integer 4 \
rttMonCtrlAdminRttType.<index> -Integer 11 \
rttMonEchoAdminProtocol.<index> -Integer 29
rttMonScheduleAdminRttStartTime.<index> -TimeTicks 1
```

6.2.12 FTP 作业

```
rttMonCtrlAdminStatus.<index> -Integer 4 \
rttMonCtrlAdminRttType.<index> -Integer 12 \
rttMonEchoAdminProtocol.<index> -Integer 30 \
rttMonEchoAdminOperation.<index> -Integer 3 \
rttMonEchoAdminURL.<index> -DisplayString
"ftp://anonymous@test:www.foo.com/foo/temp.txt" \
rttMonScheduleAdminRttStartTime.<index> -TimeTicks 1
```

6.3 关于在作业运行期间改变配置参数

由于修改配置参数会影响到已存储的数据结构，因此，在作业运行期间不能自由修改配置参数，如果需要修改配置参数，需要先将作业停止，再进行修改。

6.4 使用 SNMP 存取 SLA 的数据

不同的作业的结果数据存放在不同的表中，下面是详细介绍

6.4.1 Echo\PathEcho\UDPEcho\TCPConnect\DLSw\DNS\FTP 及 DHCP

- rttMonCtrlOperTable: 存放了最新一次的数据，每次执行完成后均更新

- rttMonStatsCaptureTable 和 rttMonStatsTotalsTable: 存储了统计数据，每次执行完成后更新。
- rttMonStatsCollect: 存储了错误统计信息，每次执行完成后更新。

6.4.2 HTTP 作业

- rttMonLatestHTTPOperTable 和 rttMonCtrlOperTable: 存放了最新一次的数据，仅 rttMonLatestRttOperCompletionTime 变量更新。
- rttMonHTTPStatsTable 和 rttMonStatsTotalsTable: 统计信息

6.4.3 抖动作业

- rttMonLatestJitterOperTable 和 rttMonCtrlOperTable: 存放了最新一次的数据，仅 rttMonLatestRttOperCompletionTime 变量更新。
- rttMonJitterStatsTable 和 rttMonStatsTotalsTable: 统计信息。

下面是对于个抖动作业的 show 的输出，其中的值已全部更换为了 SNMP 对象名：

```
#sh rtr op
Current Operational State
Entry Number: rttMonCtrlAdminIndex
Modification Time: rttMonCtrlOperModificationTime
Diagnostics Text: rttMonCtrlOperDiagText
Last Time this Entry was Reset: rttMonCtrlOperResetTime
Number of Octets in use by this Entry: rttMonCtrlOperOctetsInUse
Number of Operations Attempted: rttMonCtrlOperNumRtts
Current Seconds Left in Life: rttMonCtrlOperRttLife
Operational State of Entry: rttMonCtrlOperState
Latest Operation Start Time: rttMonLatestRttOperTime
RTT Values:
NumOfRTT: NumOfRTT       RTTSum: RTTSum  RTTSum2: RTTSum
Packet Loss Values:
PacketLossSD: PacketLossSD   PacketLossDS: PacketLossDS
PacketOutOfSequence: PacketOutOfSequence     PacketMIA: PacketMIA
PacketLateArrival: PacketTimeOut
InternalError: InternalErrorBusies: Busies
Jitter Values:
MinOfPositivesSD: MinOfPositivesSD   MaxOfPositivesSD: MaxOfPositivesSD
NumOfPositivesSD: NumOfPositivesSD   SumOfPositivesSD: SumOfPositivesSD

Sum2PositivesSD: Sum2PositivesSD
MinOfNegativesSD: MinOfNegativesSD   MaxOfNegativesSD: MaxOfNegativesSD
NumOfNegativesSD: NumOfNegativesSD   SumOfNegativesSD: SumOfNegativesSD

Sum2NegativesSD: Sum2NegativesSD
MinOfPositivesDS: MinOfPositivesDS   MaxOfPositivesDS: MaxOfPositivesDS
NumOfPositivesDS: NumOfPositivesDS   SumOfPositivesDS: SumOfPositivesDS

Sum2PositivesDS: Sum2PositivesDS
MinOfNegativesDS: MinOfNegativesDS   MaxOfNegativesDS: MaxOfNegativesDS
NumOfNegativesDS: NumOfNegativesDS   SumOfNegativesDS: SumOfNegativesDS

Sum2NegativesDS: Sum2NegativesDS
Interarrival jitterout: jitteroutrfc1989                    Interarrival
jitterin: jitterinrfc1989
```

```
One Way Values:
NumOfOW: NumOfOW
OWMinSD: OWMinSD OWMaxSD: OWMaxSD    OWSumSD: OWSumSD    OWSum2SD: OWSum2SD
OWMinDS: OWMinDS OWMaxDS: OWMaxDS    OWSumDS: OWSumDS    OWSum2DS: OWSum2DS
c36e9-3#sh rtr conf
 Complete Configuration Table (includes defaults)
Entry Number: rttMonCtrlAdminIndex
Owner: rttMonCtrlAdminOwner
Tag:
Type of Operation to Perform: rttMonCtrlAdminRttType
Reaction and History Threshold (milliseconds): rttMonCtrlAdminThreshold
Operation Frequency (seconds): rttMonCtrlAdminFrequency
Operation Timeout (milliseconds): rttMonCtrlAdminTimeout
Verify Data: rttMonCtrlAdminVerifyData
Status of Entry (SNMP RowStatus): active
Protocol Type: rttMonEchoAdminProtocol
Target Address: rttMonEchoAdminTargetAddress
Source Address: rttMonEchoAdminSourceAddress
Target Port: rttMonEchoAdminTargetPort
Source Port: rttMonEchoAdminSourcePort
Request Size (ARR data portion): rttMonEchoAdminPktDataRequestSize
Response Size (ARR data portion): rttMonEchoAdminPktDataResponseSize
Num of Packets per operation: rttMonEchoAdminNumPackets
Interval between packets(milliseconds): rttMonEchoAdminInterval
Control Packets: rttMonEchoAdminControlEnable
Loose Source Routing: rttMonEchoAdminLSREnable
LSR Path:
Type of Service Parameters: rttMonEchoAdminTOS
Life (seconds): rttMonScheduleAdminRttLife
Next Scheduled Start Time: rttMonScheduleAdminRttStartTime
Entry Ageout (seconds): rttMonScheduleAdminConceptRowAgeout
Connection Loss Reaction Enabled: rttMonReactAdminConnectionEnable
Timeout Reaction Enabled: rttMonReactAdminTimeoutEnable
Threshold Reaction Type: rttMonReactAdminThresholdType
Threshold Falling (milliseconds): rttMonReactAdminThresholdFalling
Threshold Count: rttMonReactAdminThresholdCount
Threshold Count2: rttMonReactAdminThresholdCount2
Reaction Type: rttMonReactAdminActionType
Verify Error Reaction Enabled: rttMonReactAdminVerifyErrorEnable
Number of Statistic Hours kept: rttMonStatisticsAdminNumHourGroups
Number of Statistic Paths kept: rttMonStatisticsAdminNumPaths
Number of Statistic Hops kept: rttMonStatisticsAdminNumHops
Number of Statistic Distribution Buckets kept:
rttMonStatisticsAdminNumDistBuckets
Statistic Distribution Interval (milliseconds):
rttMonStatisticsAdminDistInterval
Number of History Lives kept: rttMonHistoryAdminNumLives
Number of History Buckets kept: rttMonHistoryAdminNumBuckets
Number of History Samples kept: rttMonHistoryAdminNumSamples
History Filter Type: rttMonHistoryAdminFilter
```

6.4.4 总发包数的 OID 值

```
RttMonJitterStatsPacketMIA  1.3.6.1.4.1.9.9.42.1.3.5.1.37
RttMonJitterStatsPacketLateArrival  1.3.6.1.4.1.9.9.42.1.3.5.1.38
RttMonJitterStatsPacketLossDS   1.3.6.1.4.1.9.9.42.1.3.5.1.35
RttMonJitterStatsPacketLossSD   1.3.6.1.4.1.9.9.42.1.3.5.1.34
RttMonJitterStatsPacketOutOfSequence1.3.6.1.4.1.9.9.42.1.3.5.1.36
RttMonJitterStatsNumOfRTT   1.3.6.1.4.1.9.9.42.1.3.5.1.4
```

6.4.5 抖动作业的平均往返时间计算所需要的 OID

```
RttMonJitterStatsRTTSum                 1.3.6.1.4.1.9.9.42.1.3.5.1.5
rttMonJitterStatsNumOfRTT               1.3.6.1.4.1.9.9.42.1.3.5.1.4
```

平均 RTT＝RttMonJitterStatsRTTSum / rttMonJitterStatsNumOfRTT

6.4.6 单向度量所需要的 OID

```
RttMonJitterStatsNumOfOW 1.3.6.1.4.1.9.9.42.1.3.5.1.51
RttMonJitterStatsOWSumDS 1.3.6.1.4.1.9.9.42.1.3.5.1.41
RttMonJitterStatsOWSumSD 1.3.6.1.4.1.9.9.42.1.3.5.1.41
```

6.4.7 检查路由器内存/CPU 利用率的 OID

```
ciscoMemoryPoolFree          1.3.6.1.4.1.9.9.48.1.1.1.6
ciscoMemoryPoolUsed          1.3.6.1.4.1.9.9.48.1.1.1.5
```

内存利用率= [ciscoMemoryPoolUsed/(ciscoMemoryPoolUsed + ciscoMemoryPoolFree)] * 100

CPU 利用率：

```
usage avgBusy    5 1.3.6.1.4.1.9.2.1.58
```

系统运行时间（衡量稳定性）：

```
SysUpTime sysUpTime  1.3.6.1.2.1.1.3
```

6.4.8 历史信息

需要启动历史特性历史信息存储在 rttMonHistoryCollectionTable，当前 HTTP 和 UDP 抖动作业不支持这个表，rttMonAppl 表存储了所有和 SLA 信息，包括支持多少个作业，路由器能力等参数。

6.5 数据计算公式

可以在如下地址查找到 MIB 对象和 OID 信息：
http://www.cisco.com/pcgi-bin/Support/Mibbrowser/mibinfo.pl?mn=CISCO-RTTMON-MIB

支持如下公式：

6.5.1 带宽(Bps)

[RttMonEchoAdminNumPackets * (RttMonEchoAdminPktDataRequestSize +12) * 8] / (RttMonCtrlAdminFrequency *1000)

6.5.2 总发包数量

[RttMonJitterStatsPacketMIA + RttMonJitterStatsPacketLateArrival + RttMonJitterStatsPacketLossDS + RttMonJitterStatsPacketLossSD + RttMonJitterStatsPacketOutOfSequence + RttMonJitterStatsNumOfRTT]

6.5.3 丢包率

[(RttMonJitterStatsPacketLossDS + RttMonJitterStatsPacketLossSD + RttMonJitterStatsPacketMIA) * 100] / [RttMonJitterStatsPacketLossSD + RttMonJitterStatsPacketLossDS + RttMonJitterStatsPacketMIA + RttMonJitterStatsPacketLateArrival + RttMonJitterStatsPacketOutOfSequence + RttMonJitterStatsNumOfRTT]

6.5.4 平均往返时间(ms)

RttMonJitterStatsRTTSum / RttMonJitterStatsNumOfRTT

6.5.5 平均单向延时(ms)

源到目的:

RttMonJitterStatsOWSumSD / RttMonJitterStatsNumOfOW

目的到源：

RttMonJitterStatsOWSumDS / RttMonJitterStatsNumOfOW

6.5.6 平均抖动(ms)

源到目的:

[(RttMonStatsSumOfPosititivesSD / RttMonStatsNumOfPosititivesSD) + (RttMonStatsSumOfNegativesSD / RttMonStatsNumOfNegativesSD)] / 2

目的到源：

[(RttMonStatsSumOfPosititivesDS / RttMonStatsNumOfPosititivesDS) + (RttMonStatsSumOfNegativesDS / RttMonStatsNumOfNegativesDS)] / 2

7 NTP 相关信息

为了进行精确的单向信息测试，必须在 SLA 源路由器和目标路由器之间配置 NTP 协议，换句话说，源路由器和目的路由器必须通过 NTP 实现时钟的精确同步，需要达到时钟同步，时钟偏移最小的目标，为了达到这个目标，需要做以下几件事：

- 高品质的时钟，可以通过层级(Stratum)来识别，层级 1 优于层级 2.如果可能，建议使用 GPS 授时，GPS 授时的典型配置可参见：

  http://cco.cisco.com/en/US/products/sw/iosswrel/ps1834/products_feature_guide09186a008007fef8.html

- 将时钟源和运行 SLA 的路由器之间的跳数减到最小；
- 如果运行 SLA 的路由器到 NTP 服务器之间跳数太多，可以考虑在每个站点布署一个层 1 的 NTP 服务器；
- 如果仍然作不到，可以考虑让 SLA Responder 直接和源路由器同步。
- NTP 服务器到运行 SLA 的路由器的带宽要尽可能宽，尽可能的通过 LAN 进行时钟同步。

建议的配置命令：

```
(config)#ntp server <ntp_server_address>
```

输入完后，就会出现 ntp clock-period 一行，这个数据是根据偏移自动生成的，在复制配置是不要将它从一台路由器复制到另一台路由器上。示例如下：

```
Router# show run | include ntp
    ntp clock-period 17179759
    ntp server 10.100.71.226
```

如果条件允许，可以配置多个 NTP 服务器，示例如下：

```
Router# show run | include ntp
     ntp clock-period 17179759
     ntp server 5.1.1.32 prefer
     ntp server 10.100.71.226
     ntp server 20.200.72.39
```

NTP 配置的详细文档可以参见：

http://www.cisco.com/en/US/products/sw/iosswrel/ps1828/products_configuration_guide_chapter09186a00800ca558.html#xtocid1345017

8 加速 IP SLA 布署的网管软件简介

8.1 Cisco IPM

可以通过 Cisco IPM 加速 IP SLA（又叫作 SAA）的布署，IPM 是 CiscoWorks 2000 RWAN 中的一个重要组成部分，IPM 可以支持的 SLA 特性如下表所示：

Cisco IOS IP SLA 特性集	Cisco IPM 支持情况
ICMP Echo 作业	X
ICMP 路径 Echo 作业	X
UDP Echo 作业	X
抖动作业	X
单向抖动作业	X
HTTP 作业	X
TCP 连接作业	X
FTP 作业	X
DNS 作业	X
DLSW 作业	X
DHCP 作业	X
SNA LU2 Echo 作业	X
动作域值	X
动作域值 Trap 搜集	X
作业历史参数	X
数据分发	X
SNMP 支持	X
调度	X
帧中继作业	X
路径抖动作业	X
探知 MPLS VPN	X

应用性能监测	X
TOS 位设置	X
DSCP 位设置	X
低内存域值告警	X
MD5 控制信息	X
松散源路由	X

IPM 提供上在设备上定义 IP SLA 的模板配置，如 TCP/UDP 端口、有效荷载尺寸、包计数等。目前 IPM 未实现 trap receiver 功能，可以配置 IP SLA 启用 trap 功能。更详细的有关 IPM 的资料可以参见如下 URL：

http://www.cisco.com/en/US/products/sw/cscowork/ps1008/prod_brochure09186a0080092394.html

8.2 支持 IP SLA 的第三方网管应用

下图列出了使用了 IP SLA 特性集的第三方网管应用：

下表列出了第三方网管应用对 IP SLA 的支持情况(未列举完全)：

CISCO IOS IP SLA 特性	Quallaby Proviso 3.3	InfoVista VistaViews	Micromuse ISM	Concord E-Health
ICMP Echo 作业	X	X	X	X
ICMP 路径 Echo 作业	—	X	X	—
UDP Echo 作业	X	X	X	X
抖动作业	—	X	X	X
单向抖动作业	X	—	—	X

HTTP 作业	X	X	X	X
TCP 连接作业	X	X	X	X
FTP 作业	—	X	X	—
DNS 作业	X	X	X	X
DLSW 作业	—	X	X	—
DHCP 作业	—	X	X	X
SNA LU2 Echo 作业	—	X	X	—
动作域值	—	—	—	X
动作域值 Trap 搜集	—	—	—	X
作业历史参数	X	—	—	—
数据分发	—	—	—	—
SNMP 支持	X	X	X	X
调度	—	—	—	—
路径抖动作业	—	X	X	—
探知 MPLS VPN	—	—	—	X
应用性能监测	—	—	—	—
TOS 位设置	X	X	X	X
DSCP 位设置	X	X	X	X
低内存域值告警	—	—	—	—
MD5 控制信息	—	—	—	—
松散源路由	—	—	—	—