



Rate Based Satellite Control Protocol

Rate Based Satellite Control Protocol (RBSCP) was designed for wireless or long-distance delay links with high error rates, such as satellite links. Using tunnels, RBSCP can improve the performance of certain IP protocols, such as TCP and IP Security (IPSec), over satellite links without breaking the end-to-end model.

- [Finding Feature Information, page 1](#)
- [Restrictions for Rate Based Satellite Control Protocol, page 1](#)
- [Information About Rate Based Satellite Control Protocol, page 2](#)
- [How to Configure Rate Based Satellite Control Protocol, page 5](#)
- [Configuration Examples for Rate Based Satellite Control Protocol, page 9](#)
- [Additional References, page 11](#)
- [Feature Information for Rate Based Satellite Control Protocol, page 12](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for Rate Based Satellite Control Protocol

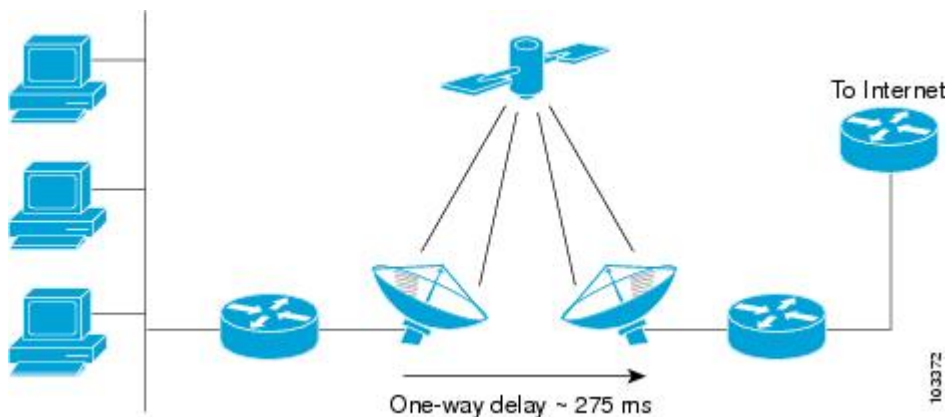
- RBSCP was designed for wireless or long-distance delay links with high error rates such as satellite links. If you do not have long-distance delay links with high error rates, do not implement this feature.
- If IP access lists (ACLs) are configured on an interface that is used by an RBSCP tunnel, the RBSCP IP protocol (199) must be allowed to enter and exit that interface or the tunnel will not function.
- RBSCP has some performance limitations because traffic through the tunnel is process-switched.

Information About Rate Based Satellite Control Protocol

IP over Satellite Links

Satellite links have several characteristics that affect the performance of IP protocols over the link. The figure below shows that satellite links can have a one-way delay of 275 milliseconds. A round-trip time (RTT) of 550 milliseconds is a very long delay for TCP. Another issue is the high error rates (packet loss rates) that are typical of satellite links as compared to wired links in LANs. Even the weather affects satellite links, causing a decrease in available bandwidth, and an increase in RTT and packet loss.

Figure 1: Typical Satellite Link



Long RTT keeps TCP in a slow start mode, which increases the time before the satellite link bandwidth is fully used. TCP and Stream Control Transmission Protocol (SCTP) interpret packet loss events as congestion in the network and start to perform congestion recovery procedures, which reduce the traffic being sent over the link.

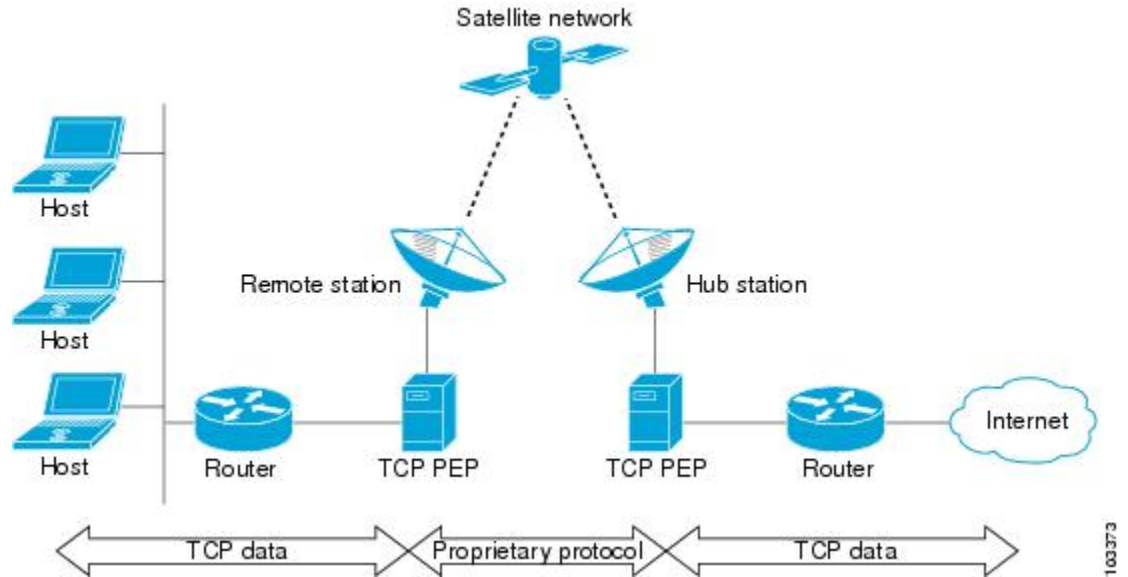
Although available satellite link bandwidths are increasing, the long RTT and high error rates experienced by IP protocols over satellite links are producing a high bandwidth-delay product (BDP).

Performance Enhancing Proxy over Satellite Links

To address the problem of TCP being kept in a slow start mode when a satellite link is used, a disruptive performance enhancing proxy (PEP) solution is introduced into the network. In the figure below you can see that the transport connection is broken up into three sections with hosts on the remote side connecting to the Internet through their default router. The router sends all Internet-bound traffic to the TCP PEP, which terminates the TCP connection to the Internet. The PEP generates a local TCP ACK (TCP spoofing) for all data. Traffic is buffered and retransmitted through a single PEP protocol connection over the satellite link. The second PEP receives the data from the satellite link and retransmits the data over separate TCP connections to the Internet. TCP transmission is disrupted, so dropped packets are not interpreted as TCP congestion and

can be retransmitted from buffered data. Minimal TCP ACKs and reduced TCP slow starts allow more bandwidth to be used.

Figure 2: Disruptive TCP PEP Solution



One of the disadvantages to using disruptive TCP PEP is the breaking of the end-to-end model. Some applications cannot work when the flow of traffic is broken, and the PEP has no provision for handling encrypted traffic (IPSec). New transport protocols such as SCTP require special handling or additional code to function with disruptive TCP PEP. An additional managed network component is also required at every satellite router.

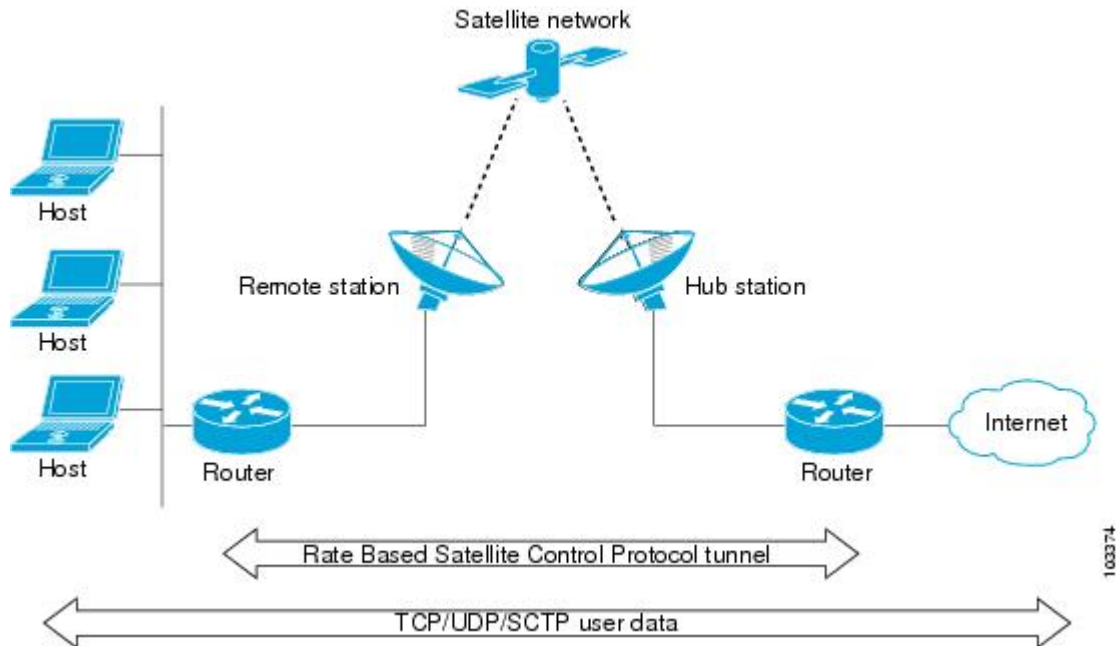
RBSCP over Satellite Links

RBSCP has been designed to preserve the end-to-end model and provide performance improvements over the satellite link without using a PEP solution. IPSec encryption of clear-text traffic (for example a VPN service configuration) across the satellite link is supported. RBSCP allows two routers to control and monitor the sending rates of the satellite link, thereby increasing the bandwidth utilization. Lost packets are retransmitted over the satellite link by RBSCP preventing the end host TCP senders from going into slow start mode.

RBSCP is implemented using a tunnel interface as shown in the figure below. The tunnel can be configured over any network interface supported by Cisco IOS software that can be used by a satellite modem or internal satellite modem network module. IP traffic is sent across the satellite link with appropriate modifications and

enhancements that are determined by the router configuration. Standard routing or policy-based routing can be used to determine the traffic to be sent through the RBSCP tunnel.

Figure 3: Nondisruptive RBSCP Solution



RBSCP tunnels can be configured for any of the following features.

Time Delay

One of the RBSCP routers can be configured to hold frames due for transmission through the RBSCP tunnel. The delay time increases the RTT at the end host and allows RBSCP time to retransmit lost TCP frames or other protocol frames. If the retransmission is successful, it prevents lost frame events from reaching the end host where congestion procedures would be enabled. In some cases the retransmission can be completed by RBSCP without inserting the delay.

ACK Splitting

Performance improvements can be made for clear-text TCP traffic using acknowledgement (ACK) splitting in which a number of additional TCP ACKs are generated for each TCP ACK received. TCP will open a congestion window by one maximum transmission unit (MTU) for each TCP ACK received. Opening the congestion window results in increased bandwidth becoming available. Configure this feature only when the satellite link is not using all the available bandwidth. Encrypted traffic cannot use ACK splitting.

Window Stuffing

Clear-text TCP and SCTP traffic can benefit from the RBSCP window stuffing feature. RBSCP can buffer traffic so that the advertised window can be incremented up to the available satellite link bandwidth or the available memory in the router. The end host that sends the packets is fooled into thinking that a larger window exists at the receiving end host and sends more traffic. Use this feature with caution because the end host may send too much traffic for the satellite link to handle and the resulting loss and retransmission of packets may cause link congestion.

SCTP Drop Reporting

SCTP uses an appropriate byte counting method instead of ACK counting to determine the size of the transmission window, so ACK splitting does not work with SCTP. The RBSCP tunnel can generate an SCTP packet-dropped report for packets dropped across the satellite but not as a result of congestion loss. This SCTP drop reporting is on by default and provides a chance to retransmit the packet without affecting the congestion window size. Actual congestion losses are still reported, and normal recovery mechanisms are activated.

How to Configure Rate Based Satellite Control Protocol

Configuring the RBSCP Tunnel

Perform this task to configure the RBSCP tunnel. Remember to configure the router at each end of the tunnel.

Before You Begin

Ensure that the physical interface to be used as the tunnel source in this task is already configured.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip unnumbered** *interface-type interface-number*
5. **tunnel source** *{ip-address | interface-type interface-number}*
6. **tunnel destination** *{hostname | ip-address}*
7. **tunnel bandwidth** *{receive | transmit} bandwidth*
8. **tunnel mode** *{aurp | cayman| dvmrp| eon| gre| gre multipoint| ipip[decapsulate-any]| iptalk| mpls| nos| rbscp}*
9. **tunnel rbscp ack_split** *split-size*
10. **tunnel rbscp delay**
11. **tunnel rbscp report**
12. **tunnel rbscp window_stuff** *step-size*
13. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> Example: Router(config)# interface tunnel 0	Specifies the interface type and number and enters interface configuration mode.
Step 4	ip unnumbered <i>interface-type interface-number</i> Example: Router(config-if)# ip unnumbered Ethernet 1	Enables IP processing on an interface without assigning an explicit IP address. <ul style="list-style-type: none"> Whenever the unnumbered interface generates a packet (for example, for a routing update), it uses the address of the specified interface as the source address of the IP packet.
Step 5	tunnel source { <i>ip-address</i> <i>interface-type interface-number</i> } Example: Router(config-if)# tunnel source Ethernet 1	Configures the tunnel source. <ul style="list-style-type: none"> Use the <i>ip-address</i> argument to specify the IP address of the service provider. Use the <i>interface-type</i> and <i>interface-number</i> arguments to specify the interface to use. For RBSCP we recommend specifying an interface as the tunnel source.
Step 6	tunnel destination { <i>hostname</i> <i>ip-address</i> } Example: Router(config-if)# tunnel destination 172.17.2.1	Configures the tunnel destination. <ul style="list-style-type: none"> Use the <i>hostname</i> argument to specify the IP address of the service provider. Use the <i>ip-address</i> argument to specify the interface to use.
Step 7	tunnel bandwidth { <i>receive</i> <i>transmit</i> } <i>bandwidth</i> Example: Router(config-if)# tunnel bandwidth transmit 1000	Specifies the tunnel bandwidth to be used to transmit packets. <ul style="list-style-type: none"> Use the <i>bandwidth</i> argument to specify the bandwidth. <p>Note The receive keyword is no longer used.</p>
Step 8	tunnel mode { <i>aurp</i> <i>cayman</i> <i>dvmrp</i> <i>eon</i> <i>gre</i> <i>gre multipoint</i> <i>ipip</i> [<i>decapsulate-any</i>] <i>iptalk</i> <i>mpls</i> <i>nos</i> <i>rbscp</i> } Example: Router(config-if)# tunnel mode rbscp	Specifies the protocol to be used in the tunnel. <ul style="list-style-type: none"> Use the rbscp keyword to specify that RBSCP will be used as the tunnel protocol.

	Command or Action	Purpose
Step 9	tunnel rbscp ack_split <i>split-size</i> Example: <pre>Router(config-if)# tunnel rbscp ack_split 6</pre>	(Optional) Enables TCP acknowledgement (ACK) splitting with RBSCP tunnels. <ul style="list-style-type: none"> • Use the <i>split-size</i> argument to specify the number of ACKs to send for every ACK received. • The default number of ACKs is 4.
Step 10	tunnel rbscp delay Example: <pre>Router(config-if)# tunnel rbscp delay</pre>	(Optional) Enables RBSCP tunnel delay. <ul style="list-style-type: none"> • Use this command only when the RTT is greater than 700 milliseconds.
Step 11	tunnel rbscp report Example: <pre>Router(config-if)# tunnel rbscp report</pre>	(Optional) Reports dropped RBSCP packets to SCTP. <ul style="list-style-type: none"> • Reporting dropped packets to SCTP provides better bandwidth use because SCTP tells the end hosts to retransmit the dropped packets and this prevents the end hosts from assuming that the network is congested.
Step 12	tunnel rbscp window_stuff <i>step-size</i> Example: <pre>Router(config-if)# tunnel rbscp window_stuff 3</pre>	(Optional) Enables TCP window stuffing by increasing the value of the TCP window scale for RBSCP tunnels. <ul style="list-style-type: none"> • Use the <i>step-size</i> argument to specify the step increment number.
Step 13	exit Example: <pre>Router(config-if)# exit</pre>	Exits interface configuration mode and returns to global configuration mode. <ul style="list-style-type: none"> • Repeat this step one more time to exit global configuration mode.

What to Do Next

This task must be repeated on the router on the other side of the satellite link. Substitute the sample IP addresses, host names, and other parameters for the appropriate values on the second router.

Verifying that the RBSCP Tunnel Is Active

Perform this task to verify that the RBSCP tunnel configured in the [Configuring the RBSCP Tunnel](#), on page 5 task is active.

SUMMARY STEPS

1. **enable**
2. **show rbscp [all| state| statistics] [tunnel *tunnel-number*]**
3. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show rbscp [all state statistics] [tunnel <i>tunnel-number</i>] Example: Router# show rbscp state tunnel 1	Use this command with the state and tunnel keywords to display information about the current state of the tunnel. In the following sample output the tunnel is shown in an open state. All counters display totals accumulated since the last clear rbscp command was issued.
Step 3	exit Example: Router># exit	Exits privileged EXEC configuration mode.

Verifying the RBSCP Traffic

Perform this task to verify that the traffic is being transmitted through the RBSCP tunnel and across the satellite link.

SUMMARY STEPS

1. **enable**
2. **show interfaces tunnel *number* [accounting]**
3. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Router> enable	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show interfaces tunnel <i>number</i> [accounting] Example: Router# show interfaces tunnel 0	Use this command to show that traffic is being transmitted through the RBSCP tunnel.
Step 3	exit Example: Router># exit	Exits privileged EXEC configuration mode.

Configuration Examples for Rate Based Satellite Control Protocol

Configuring the RBSCP Tunnel Example

In the following example, Router 1 and Router 2 are configured to send traffic through an RBSCP tunnel over a satellite link.

Router 1

```
interface Tunnel 0
 ip unnumbered ethernet1
 tunnel source ethernet1
 tunnel destination 172.17.2.1
 tunnel bandwidth transmit 1000
 tunnel mode rbscp
 tunnel rbscp ack_split 6
 tunnel rbscp report
!
interface ethernet1
 description Satellite Link
 ip address 172.20.1.2 255.255.255.0
```

Router 2

```
interface Tunnel 0
 ip unnumbered ethernet1
 tunnel source ethernet1
 tunnel destination 172.20.1.2
 tunnel bandwidth transmit 1000
```

```

tunnel mode rbscp
tunnel rbscp ack_split 6
tunnel rbscp report
!
interface ethernet1
description Satellite Link
ip address 172.17.2.1 255.255.255.0

```

Configuring Routing for the RBSCP Tunnel Example

To control the type of traffic that uses the RBSCP tunnel, you must configure the appropriate routing. If you want to direct all traffic through the tunnel, you can configure a static route.



Note

Remember to configure the tunnel interface as passive if dynamic routing protocols are used to prevent routing flaps.

The following example shows how to use policy based routing to route some specific protocol types through the tunnel. In this example, an extended access list allows TCP, Stream Control Transmission Protocol (SCTP), Encapsulating Security Payload (ESP) protocol, and Authentication Header (AH) traffic to travel through the tunnel. All IP traffic is denied.

Router 1 (Local Side)

```

interface Tunnell
ip unnumbered FastEthernet1/1
tunnel source FastEthernet1/1
tunnel destination 10.12.0.20
tunnel mode rbscp
tunnel ttl 5
tunnel bandwidth transmit 30000
tunnel rbscp window_stuff 1
tunnel rbscp ack_split 4
!
interface FastEthernet0/0
ip address 10.13.0.1 255.255.255.0
ip policy route-map rbscp-pbr
duplex auto
speed auto
!
interface FastEthernet1/1
description Satellite Link
ip address 10.12.0.1 255.255.255.0
duplex auto
speed auto
!
ip route 10.15.0.0 255.255.255.0 FastEthernet1/1
!
ip access-list extended rbscp-acl
permit tcp any 10.15.0.0 0.0.0.255
permit 132 any 10.15.0.0 0.0.0.255
permit esp any 10.15.0.0 0.0.0.255
permit ahp any 10.15.0.0 0.0.0.255
deny ip any any
!
!
route-map rbscp-pbr permit 10
match ip address rbscp-acl
set interface Tunnell

```

Router 2 (Remote Side)

```

ip dhcp pool CLIENT
  import all
  network 10.15.0.0 255.255.255.0
  default-router 10.15.0.1
  domain-name engineer.chicago.il.us
  dns-server 10.10.0.252
!
interface Tunnell
  ip unnumbered FastEthernet0/1
  tunnel source FastEthernet0/1
  tunnel destination 10.12.0.1
  tunnel mode rbscp
  tunnel ttl 5
  tunnel bandwidth transmit 30000
  tunnel rbscp window_stuff 1
  tunnel rbscp ack_split 4
!
interface FastEthernet0/0
  description Local LAN
  ip address 10.15.0.1 255.255.255.0
  ip policy route-map rbscp-pbr
  duplex auto
  speed auto
!
interface FastEthernet0/1
  description Satellite Link
  ip address 10.12.0.20 255.255.255.0
  duplex auto
  speed auto
!
ip route 0.0.0.0 0.0.0.0 FastEthernet0/1
!
ip access-list extended rbscp-acl
  permit tcp any any
  permit 132 any any
  permit esp any any
  permit ahp any any
  deny ip any any
!
route-map rbscp-pbr permit 10
  match ip address rbscp-acl
  set interface Tunnell

```

Additional References

Related Documents

Related Topic	Document Title
Tunnel interface commands: complete command syntax, command mode, defaults, usage guidelines, and examples	Cisco IOS Interface and Hardware Component Command Reference
Tunnel configuration	Cisco IOS Interface and Hardware Component Configuration Guide

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	--

MIBs

MIBs	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 1323	<i>TCP Extensions for High Performance</i>
RFC 2018	<i>TCP Selective Acknowledgment Options</i>

Technical Assistance

Description	Link
Technical Assistance Center (TAC) home page, containing 30,000 pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/public/support/tac/home.shtml

Feature Information for Rate Based Satellite Control Protocol

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [http://www.cisco.com/go/featurenavigator](#). An account on Cisco.com is not required.

Table 1: Feature Information for AToM NSF Any Transport over MPLS and AToM Graceful Restart

Feature Name	Releases	Feature Information
Rate Based Satellite Control Protocol	12.3(7)T	This feature was introduced. The following commands were introduced or modified: clear rbscp, debug tunnel rbscp, show rbscp, tunnel bandwidth, tunnel mode, tunnel rbscp ack_split, tunnel rbscp delay, tunnel rbscp input_drop, tunnel rbscp long_drop, tunnel rbscp report, tunnel rbscp, window_stuff.

