



Interface and Hardware Component Configuration Guide, Cisco IOS Release 15SY

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

© 2012 Cisco Systems, Inc. All rights reserved.



CONTENTS

IP over IPv6 Tunnels 1

- Finding Feature Information 1
- Information About IP over IPv6 Tunnels 2
 - Overlay Tunnels for IPv6 2
 - IPv6 Manually Configured Tunnels 4
 - GRE IPv4 Tunnel Support for IPv6 Traffic 4
 - GRE Support over IPv6 Transport 4
- How to Configure IP over IPv6 Tunnels 5
 - Configuring Manual IPv6 Tunnels 5
 - Configuring GRE on IPv6 Tunnels 7
- Configuration Examples for IP over IPv6 Tunnels 8
 - Example: Configuring Manual IPv6 Tunnels 9
 - Example: Configuring GRE on IPv6 Tunnels 9
- Additional References 9
- Feature Information for IP over IPv6 Tunnels 10

Configuring LAN Interfaces 13

- Finding Feature Information 13
- Configuring Ethernet Fast Ethernet or Gigabit Ethernet Interfaces 14
 - Ethernet Fast Ethernet and Gigabit Ethernet Interface Configuration Task List 15
 - Specifying an Ethernet Fast Ethernet or Gigabit Ethernet Interface 15
 - Specifying an Ethernet Encapsulation Method 16
 - Specifying Full-Duplex Operation 16
 - Specifying the Media and Connector Type 17
 - Extending the 10BASE-T Capability 18
 - Configuring Fast Ethernet 100BASE-T 18
 - Configuring PA-12E 2FE Port Adapters 20
 - Configuring the PA-12E 2FE Port Adapter 21
 - Configuring the PA-12E 2FE Port Adapter 22
 - Monitoring and Maintaining the PA-12E 2FE Port Adapter 23

Configuring Bridge Groups Using the 12E 2FE VLAN Configuration WebTool	25
Configuring the 100VG-AnyLAN Port Adapter	26
Configuring the Cisco 7200-I O-GE+E and Cisco 7200-I O-2FE E Input Output Controllers	27
Cisco 7200-I O-GE+E and Cisco 7200-I O-2FE E Configuration Task List	27
Configuring the Interface Transmission Mode	28
Configuring Interface Speed	28
Configuring the Ethernet Fast Ethernet and Gigabit Ethernet Interfaces	29
Verifying the Configuration	30
Monitoring and Maintaining the Cisco 7200-I O GE+E and Cisco 7200-I O-2FE E	31
Configuring Fast EtherChannel	32
Fast EtherChannel Configuration Task List	33
Configuring the Port-Channel Interface	33
Configuring the Fast Ethernet Interfaces	34
Configuring the Fast Ethernet Interfaces	35
Configuring the Gigabit Ethernet Interfaces	37
Configuring the Gigabit Ethernet Interfaces	37
Configuring a FDDI Interface	38
Source-Route Bridging over FDDI on Cisco 4000-M Cisco 4500-M and Cisco 4700-M Routers	39
Particle-Based Switching of Source-Route Bridge Packets on Cisco 7200 Series Routers	39
Using Connection Management Information	39
FDDI Configuration Task List	40
Specifying a FDDI Interface	41
Enabling FDDI Bridging Encapsulation	41
Enabling Full-Duplex Mode on the FDDI Interface	41
Setting the Token Rotation Time	42
Setting the Transmission Valid Timer	42
Controlling the Transmission Timer	42
Modifying the C-Min Timer	43
Modifying the TB-Min Timer	43
Modifying the FDDI Timeout Timer	43
Controlling SMT Frame Processing	43
Enabling Duplicate Address Checking	43
Setting the Bit Control	44
Controlling the CMT Microcode	44

Starting and Stopping FDDI	44
Setting FDDI Frames per Token Limit	45
Controlling the FDDI SMT Message Queue Size	45
Preallocating Buffers for Bursty FDDI Traffic	45
Configuring a Hub Interface	46
Enabling a Hub Port	46
Disabling or Enabling Automatic Receiver Polarity Reversal	46
Disabling or Enabling the Link Test Function	47
Enabling Source Address Control	47
Enabling SNMP Illegal Address Trap	48
Configuring a Token Ring Interface	48
Particle-Based Switching of Source-Route Bridge Packets on Cisco 7200 Series Routers	49
Dedicated Token Ring Port Adapter	49
Token Ring Interface Configuration Task List	50
Specifying a Token Ring Interface	50
Enabling Early Token Release	50
Configuring PCbus Token Ring Interface Management	50
Enabling a Token Ring Concentrator Port	51
Monitoring and Maintaining the Port	51
LAN Interface Configuration Examples	51
Ethernet Encapsulation Enablement Example	51
Full-Duplex Enablement Operation Example	52
PA-12E 2FE Port Configuration Examples	52
PA-VG100 Port Adapter Configuration Example	53
Cisco 7200-I O-GE+E and Cisco 7200-I O-2FE E Configuration Examples	53
Configuring the Gigabit Ethernet Interface on the Cisco 7200-I O-GE+E	54
Configuring Autonegotiation on the Cisco 7200-I O-2FE E	54
Fast EtherChannel Configuration Examples	54
FDDI Frames Configuration Example	56
Hub Configuration Examples	56
Hub Port Startup Examples	56
Source Address for an Ethernet Hub Port Configuration Examples	56
Hub Port Shutdown Examples	57
SNMP Illegal Address Trap Enablement for Hub Port Example	57
Configuring Serial Interfaces	59

Finding Feature Information	59
Prerequisites for Configuring Serial Interfaces	60
Restrictions for Configuring Serial Interfaces	60
Information About Configuring Serial Interfaces	60
Cisco HDLC Encapsulation	60
PPP Encapsulation	61
Multilink PPP	61
Keepalive Timer	62
Frame Relay Encapsulation	62
LMI on Frame Relay Interfaces	63
Layer 2 Tunnel Protocol Version 3-Based Layer 2 VPN on Frame Relay	64
High-Speed Serial Interfaces	64
How to Configure Serial Interfaces	65
Configuring a High-Speed Serial Interface	65
Specifying a HSSI Interface	65
Specifying HSSI Encapsulation	65
Invoking ATM on a HSSI Line	66
Converting HSSI to Clock Master	66
Configuring a Synchronous Serial Interface	66
Synchronous Serial Configuration Task List	67
Specifying a Synchronous Serial Interface	67
Specifying Synchronous Serial Encapsulation	68
Configuring PPP	69
Configuring Half-Duplex and Bisync for Synchronous Serial Port Adapters on Cisco 7200 Series Routers	69
Configuring Bisync	69
Configuring Half-Duplex Carrier Modes and Timers	70
Configuring Compression Service Adapters on Cisco 7500 Series Routers	70
Configuring Compression of HDLC Data	70
Configuring Real-Time Transport Protocol Header Compression	71
Configuring the Data Compression AIM	71
Configuring PPP Compression	72
Verifying PPP Compression	73
Configuring Frame Relay Map Compression	73
Verifying Frame Relay Map Compression	74

Configuring Frame Relay Payload Compression	76
Verifying Frame Relay Payload Compression	77
Configuring Diagnostics	78
Verifying Diagnostics	78
Configuring the CRC	79
Using the NRZI Line-Coding Format	80
Enabling the Internal Clock	80
Inverting the Data	80
Inverting the Transmit Clock Signal	81
Setting Transmit Delay	81
Configuring DTR Signal Pulsing	81
Ignoring DCD and Monitoring DSR as Line Up Down Indicator	82
Specifying the Serial Network Interface Module Timing	82
Specifying G.703 and E1-G.703 G.704 Interface Options	83
Enabling Framed Mode	83
Enabling CRC4 Generation	83
Using Time Slot 16 for Data	84
Specifying a Clock Source	84
Configuring a Channelized T3 Interface Processor	84
Channelized T3 Configuration Task List	85
Configuring T3 Controller Support for the Cisco AS5800	86
Configuring the T3 Controller	88
Configuring Each T1 Channel	89
Configuring External T1 Channels	91
Monitoring and Maintaining the CT3IP	92
Verifying T3 Configuration	93
Configuring Maintenance Data Link Messages	94
Enabling Performance Report Monitoring	94
Configuring for BERT on the Cisco AS5300	95
Verifying BERT on the Cisco AS5300	96
Enabling a BERT Test Pattern	96
Enabling Remote FDL Loopbacks	97
Configuring T1 Cable Length and T1 E1 Line Termination	98
Configuring PA-E3 and PA-2E3 Serial Port Adapters	100
PA-E3 and PA-2E3 Serial Port Adapter Configuration Task List	100

Configuring the PA-E3 Port Adapter	101
Monitoring and Maintaining the PA-E3 Port Adapter	102
Configuring PA-T3 and PA-2T3 Serial Port Adapters	102
PA-T3 and PA-2T3 Port Adapter Configuration Task List	103
Configuring the PA-T3 Port Adapter	103
Troubleshooting the PA-T3 Port Adapter	105
Monitoring and Maintaining the PA-T3 Port Adapter	105
Configuring a Packet OC-3 Interface	106
Packet OC-3 Interface Configuration Task List	107
Selecting a Packet OC-3 Interface	108
Setting the MTU Size	108
Configuring Framing	108
Configuring an Interface for Internal Loopback	109
Configuring an Interface for Line Loopback	109
Setting the Source of the Transmit Clock	109
Enabling Payload Scrambling	109
Configuring an Alarm Indication Signal	110
Configuring a DPT OC-12c Interface	110
OC-12c Interface Configuration Task List	111
Configuring the Dynamic Packet Transport Interface	111
Configuring Intelligent Protection Switching	112
Configuring DPT Topology	113
Configuring Automatic Protection Switching of Packet-over-SONET Circuits	114
APS Configuration Task List	115
Configuring APS Working and Protect Interfaces	115
Configuring APS Working and Protect Interfaces	116
Configuring Other APS Options	117
Monitoring and Maintaining APS	118
Configuring SONET Alarm Reporting	118
Configuring a Protection Switch	118
Configuring Serial Interfaces for CSU DSU Service Modules	119
Fractional T1 FT WIC CSU DSU Service Module Configuration Task List	119
Specifying the Clock Source	120
Enabling Data Inversion Before Transmission	120
Specifying the Frame Type of an FT T1 Line	120

Specifying the CSU Line Build-Out	121
Specifying FT1 T1 Line-Code Type	121
Enabling Remote Alarms	121
Enabling Loop Codes That Initiate Remote Loopbacks	122
Specifying Time Slots	123
Enabling the T1 CSU WIC	123
2-Wire and 4-Wire 56 64-kbps CSU DSU Service Module Configuration Task List	124
Setting the Clock Source	124
Setting the Network Line Speed	125
Enabling Scrambled Data Coding	125
Changing Between Digital Data Service and Switched Dial-Up Modes	126
Enabling Acceptance of a Remote Loopback Request	126
Selecting a Service Provider	127
Configuring Low-Speed Serial Interfaces	127
Understanding Half-Duplex DTE and DCE State Machines	127
Half-Duplex DTE State Machines	128
Half-Duplex DCE State Machines	129
Changing Between Controlled-Carrier and Constant-Carrier Modes	131
Placing a Low-Speed Serial Interface in Controlled-Carrier Mode	131
Placing a Low-Speed Serial Interface in Constant-Carrier Mode	132
Tuning Half-Duplex Timers	132
Changing Between Synchronous and Asynchronous Modes	133
Automatic Removal of tftp ftp rcp Source Interfaces Configuration	133
Troubleshooting Serial Interfaces	134
Troubleshooting Channelized T1 or E1	134
Running Controller Loopback Diagnostic Tests	134
Local Loopback	135
Remote Loopback	135
Channelized E1 Controller Loopback	136
Troubleshooting the T3 and T1 Channels on the CT3IP	137
Enabling Test Port	137
Enabling Test Port	138
Loopback T1 Channels	138
Loopback T3 Lines	141
Troubleshooting the PA-E3 Port Adapter	142

Configuration Examples for Serial Interface Configuration	142
Interface Enablement Configuration Examples	143
HSSI Configuration Examples	143
Channelized T3 Interface Processor Configuration Examples	143
Typical CT3IP Controller Configuration Examples	144
CT3IP Configuration with Default Values Accepted Example	145
CT3IP External Ports Configuration Example	145
CT3IP Maintenance Data Link Example	145
CT3IP Performance Monitoring Example	145
BERT Profile Configuration Example	146
E2 Clock Rate Configuration Example	146
CT3IP BERT Test Pattern Example	146
CT3IP Remote FDL Loopback Example	146
PA-E3 Serial Port Adapter Configuration Example	146
PA-T3 and PA-2T3 Configuration Example	147
Packet OC-3 Interface Configuration Examples	147
Packet-Over-SONET OC-3 Configuration Example	147
Packet OC-3 Configuration with Default Values Accepted Example	147
Two Routers Connected Back-to-Back Example	147
DPT OC-12c Interface Configuration Examples	148
DPT Port Adapter Configuration Example	148
DPT Interface Processor Configuration Example	148
IPS Options Configuration Example	148
DPT Topology Configuration Example	148
APS Configuration Examples	149
Basic APS Configuration Example	149
Multiple APS Interfaces Configuration Example	150
CSU DSU Service Module Examples	151
FT1 T1 Examples	151
T1 Frame Type Example	151
CSU Line Build-Out Example	151
T1 Line-Code Type Example	151
Loop Codes Example	151
Time Slots Example	152
Performance Report Example	152

Loopback Line Enablement Examples	152
Loopback DTE Examples	152
Clock Source Example	153
TI CSU WIC Configuration Example	153
2- and 4-Wire 56 64-kbps Service Module Examples	153
Network Line Speed Examples	153
Scrambled Data Coding Example	154
Switched Dial-Up Mode Example	154
Performance Report Example	154
Remote Loopback Request Example	154
Service Provider Example	155
E1-G.703 G.704 Serial Port Adapter Example	155
Low-Speed Serial Interface Examples	155
Synchronous or Asynchronous Mode Examples	155
Controlled-Carrier and Constant-Carrier Mode Examples	156
Half-Duplex Timers Example	156
Cisco 4000 Series Router with 2T16S Serial Network Processor Examples	156
Additional References	160
Feature Information for Configuring Serial Interfaces	161
Circuit Emulation over IP	163
Finding Feature Information	163
Prerequisites for Circuit Emulation over IP	163
Restrictions for Circuit Emulation over IP	163
Information About Circuit Emulation over IP	164
Circuit Emulation over IP	164
Benefits of CEM over IP	165
Adaptive Clocking for CEoIP	165
Clock Switchover	165
Payload Compression for CEoIP	165
Data Protection (Sample Repetition)	165
Dejitter	165
Idle Pattern	166
Payload Size	166
Signaling for CEoIP	166
Control Lead Configurations	166

How to Configure Circuit Emulation over IP	166
Configuring the NM-CEM-4TE1 Card Type	166
What to Do Next	167
Configuring the T1 E1 Line	167
What to Do Next	170
Creating CEM Channels on the T1 E1 Line	170
What to Do Next	171
Configuring the Connection Using the xconnect Command	171
What to Do Next	173
Configuring the CEM Channel	173
What to Do Next	179
Configuration Examples for CEoIP	179
Configuring a T1 CEM Network Module Example	179
Additional References	180
Command Reference	181
Feature Information for Circuit Emulation over IP	182
Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source	183
Finding Feature Information	184
Prerequisites for Cisco Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source	184
Restrictions for Cisco Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source	184
Information About Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source	185
Lossless Compression Codec on NM-HDV	186
ATM Cell Switching on AIM-ATM and AIM-ATM-VOICE-30	186
BITS Clocking on the Cisco 3660 and Cisco 3745	186
How to Configure Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source	186
Configuring the Cell Site Router for BITS Clocking	187
Configuring ATM Cell Switching	188
Configuring the Lossless Compression Codec	191
Disabling Connection Admission Control	196
Detailed Steps	197
Verifying Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source Configuration	198

Additional References	200
Feature Information for Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source	201
Finding Feature Information	203
Minimal Disruptive Restart of VIP Cards	205
Rate Based Satellite Control Protocol	209
Finding Feature Information	209
Restrictions for Rate Based Satellite Control Protocol	209
Information About Rate Based Satellite Control Protocol	209
IP over Satellite Links	210
Performance Enhancing Proxy over Satellite Links	210
RBSCP over Satellite Links	211
How to Configure Rate Based Satellite Control Protocol	213
Configuring the RBSCP Tunnel	213
What to Do Next	216
Verifying that the RBSCP Tunnel Is Active	216
Verifying the RBSCP Traffic	217
Configuration Examples for Rate Based Satellite Control Protocol	217
Configuring the RBSCP Tunnel Example	218
Configuring Routing for the RBSCP Tunnel Example	218
Additional References	220
Feature Information for Rate Based Satellite Control Protocol	221
Configuring Virtual Interfaces	223
Finding Feature Information	223
Prerequisites for Configuring Virtual Interfaces	223
Information About Configuring Virtual Interfaces	224
Virtual Interfaces	224
Benefits of Virtual Interfaces	224
Loopback Interfaces	225
Loopback Interfaces Versus Loopback Mode	226
Null Interfaces	226
Subinterfaces	227
Tunnel Interfaces	227
Virtual Multipoint Interface	228
How to Configure Virtual Interfaces	228
Configuring a Loopback Interface	228

Configuring a Null Interface	230
ICMP Unreachable Messages from Null Interfaces	230
Configuring a Subinterface	232
Configuration Examples for Virtual Interfaces	234
Configuring a Loopback Interface Example	234
Configuring a Null Interface Example	234
Configuring a Subinterface Example	234
Where to Go Next	234
Additional References	235
Feature Information for Configuring Virtual Interfaces	236
Implementing Tunnels	239
Finding Feature Information	239
Prerequisites for Implementing Tunnels	239
Restrictions for Implementing Tunnels	240
Information About Implementing Tunnels	241
Tunneling Versus Encapsulation	241
Definition of Tunneling Types by OSI Layer	242
Benefits of Tunneling	246
Tunnel ToS	247
Mobile IP Tunneling	247
Generic Routing Encapsulation	248
GRE Tunnel IP Source and Destination VRF Membership	248
EoMPLS over GRE	248
Multipoint GRE Tunneling	250
GRE CLNS Tunnel Support for IPv4 and IPv6 Packets	250
GRE IPv4 Tunnel Support for IPv6 Traffic	250
Overlay Tunnels for IPv6	251
IPv6 Manually Configured Tunnels	253
Automatic 6to4 Tunnels	253
Automatic IPv4-Compatible IPv6 Tunnels	254
ISATAP Tunnels	254
Rate-Based Satellite Control Protocol Tunnels	255
Path MTU Discovery	258
QoS Options for Tunnels	258
How to Implement Tunnels	259

Determining the Tunnel Type	259
What to Do Next	260
Configuring a GRE Tunnel	261
GRE Tunnel Keepalive	261
What to Do Next	264
Configuring GRE IPv6 Tunnels	264
What to Do Next	266
Configuring GRE Tunnel IP Source and Destination VRF Membership	266
What to Do Next	268
Configuring a CTunnel	268
CTunnel	268
Troubleshooting Tips	270
What to Do Next	270
Configuring GRE CLNS CTunnels to Carry IPv4 and IPv6 Packets	270
Tunnels for IPv4 and IPv6 Packets over CLNS Networks	270
What to Do Next	272
Configuring Manual IPv6 Tunnels	272
What to Do Next	274
Configuring 6to4 Tunnels	274
What to Do Next	277
Configuring IPv4-Compatible IPv6 Tunnels	277
What to Do Next	278
Configuring ISATAP Tunnels	278
What to Do Next	280
Configuring the RBSCP Tunnel	280
What to Do Next	283
Verifying Tunnel Configuration and Operation	283
Verifying RBSCP Tunnel Configuration and Operation	285
Verifying That the RBSCP Tunnel Is Active	285
Verifying the RBSCP Traffic	286
Configuration Examples for Implementing Tunnels	288
Example Configuring GRE IPv4 Tunnels	288
Example: Configuring GRE IPv6 Tunnels	289
Example Configuring GRE Tunnel IP Source and Destination VRF Membership	290
Example Routing Two AppleTalk Networks Across an IP-Only Backbone	291

Example Routing a Private IP Network and a Novell Network Across a Public Service Provider	292
Example Configuring a CTunnel	293
Example Configuring GRE CLNS CTunnels to Carry IPv4 and IPv6 Packets	294
Example Configuring Manual IPv6 Tunnels	295
Example Configuring 6to4 Tunnels	295
Example Configuring IPv4-Compatible IPv6 Tunnels	296
Example Configuring ISATAP Tunnels	296
Example Configuring the RBSCP Tunnel	297
Example Configuring Routing for the RBSCP Tunnel	297
Example Configuring QoS Options on Tunnel Interfaces	299
Example Configuring EoMPLS over GRE	300
Where to Go Next	301
Additional References	302
Feature Information for Implementing Tunnels	304
Tunnel Route Selection	309
Finding Feature Information	309
Prerequisites for Tunnel Route Selection	309
Restrictions for Tunnel Route Selection	309
Information About Tunnel Route Selection	310
Tunnel Transport Behavior	310
How to Configure Tunnel Route Selection	311
Configuring Tunnel Route Selection	311
Troubleshooting Tips	312
What to Do Next	312
Configuration Examples for Tunnel Route Selection	312
Example Configuring Tunnel Route Selection	313
Additional References	313
Feature Information for Tunnel Route Selection	314
MPLS VPN over mGRE	317
Finding Feature Information	317
Prerequisites for MPLS VPN over mGRE	317
Restrictions for MPLS VPN over mGRE	318
Information About MPLS VPN over mGRE	318
MPLS VPN over mGRE	318

Route Maps	319
Tunnel Endpoint Discovery and Forwarding	319
Tunnel Decapsulation	320
Tunnel Source	320
IPv6 VPN	320
How to Configure MPLS VPN over mGRE	320
Configuring an L3VPN Encapsulation Profile	320
Configuring BGP and Route Maps	322
Configuration Examples for MPLS VPN over mGRE	328
Example Verifying The MPLS VPN over mGRE Configuration	328
Example Configuration Sequence For MPLS VPN over mGRE	328
Additional References	330
Feature Information for MPLS VPN over mGRE	331
IP Tunnel MIBs	333
Finding Feature Information	333
Prerequisites for the IP Tunnel MIB	333
Restrictions for the IP Tunnel MIB	333
Information About the IP Tunnel MIB	334
Benefits of the IP Tunnel MIB	334
MIB Objects Supported by the IP Tunnel MIB	334
How to Configure SNMP and Use the IP Tunnel MIB	336
Configuring the Router to Use SNMP	336
What to Do Next	337
Additional References	337
Feature Information for the Tunnel MIB	339
Managing Dial Shelves	341
Finding Feature Information	341
Dial Shelf Management Task List	341
Understanding Shelf Architecture and DSIP	342
Maintaining Shelves	342
Configuring the Shelf IDs	342
Configuring the Shelf IDs	344
Executing Commands Remotely	345
Maintaining Dial Shelf Controllers	345
Configuring Clocks	346

Monitoring and Maintaining DSCs and Feature Boards	346
Troubleshooting Using DSIP	347
Router-Shelf Redundancy for the Cisco AS5800	351
Finding Feature Information	351
Feature Overview	351
Additional Considerations	352
System Controller	352
Load Sharing	352
Hitless Redundancy	353
Network Management	353
Benefits	353
Restrictions	353
Related Features and Technologies	354
Prerequisites	354
Configuration Examples	354
Configuring the Cisco AS5800 for Shelf Redundancy	354
Configuring the Shelf Redundancy Feature	355
Verifying Shelf Redundancy	355
Additional References	357
Feature Information for Router-Shelf Redundancy for the Cisco AS5800	358
Glossary	358
Route-Switch-Controller Handover Redundancy on the Cisco AS5850	361
Finding Feature Information	361
Feature Overview	362
Benefits	362
Restrictions	363
Related Features and Technologies	363
Related Documents	363
Supported Platforms	364
Supported Standards MIBs and RFCs	364
Prerequisites	365
Configuration Tasks	365
Configuring Classic-Split Mode	366
Verifying Classic-Split Mode	367
Configuring Handover-Split Mode	368

Configuring Handover-Split Mode	368
Verifying Handover-Split Mode	369
Troubleshooting Tips	372
Monitoring and Maintaining Handover Redundancy	372
Configuration Examples	373
Command Reference	379
Glossary	379
Route Processor Redundancy Plus (RPR+)	381
Finding Feature Information	381
Prerequisites for Route Processor Redundancy Plus (RPR+)	381
Restrictions for Route Processor Redundancy Plus (RPR+)	382
Information About Route Processor Redundancy Plus (RPR+)	382
RPR	382
RPR+	382
How to Configure Route Processor Redundancy Plus (RPR+)	383
Copying an Image onto Active and Standby RSPs	383
What to Do Next	385
Setting the Configuration Register Boot Variable	385
Configuring RPR+	387
Verifying RPR+	389
Configuration Examples for Route Processor Redundancy Plus (RPR+)	390
Configuring RPR+ Example	390
Additional References	391
Feature Information for Route Processor Redundancy Plus (RPR+)	393
Glossary	393
IPv6 Automatic 6to4 Tunnels	395
Finding Feature Information	395
Information About IPv6 Automatic 6to4 Tunnels	395
Automatic 6to4 Tunnels	395
How to Configure IPv6 Automatic 6to4 Tunnels	396
Configuring Automatic 6to4 Tunnels	396
Configuration Examples for IPv6 Automatic 6to4 Tunnels	398
Example: Configuring 6to4 Tunnels	398
Additional References	398
Feature Information for IPv6 Automatic 6to4 Tunnels	399

IPv6 Automatic IPv4-Compatible Tunnels	401
Finding Feature Information	401
Information About IPv6 Automatic IPv4-Compatible Tunnels	401
Overlay Tunnels for IPv6	401
Automatic IPv4-Compatible IPv6 Tunnels	404
How to Configure IPv6 Automatic IPv4-Compatible Tunnels	404
Configuring IPv4-Compatible IPv6 Tunnels	404
Configuration Examples for IPv6 Automatic IPv4-Compatible Tunnels	405
Example: Configuring IPv4-Compatible IPv6 Tunnels	405
Additional References	406
Feature Information for IPv6 Automatic IPv4-Compatible Tunnels	407
IPv6 GRE Tunnels in CLNS Networks	409
Finding Feature Information	409
Information About IPv6 GRE Tunnels in CLNS Networks	409
Overlay Tunnels for IPv6	409
GRE CLNS Tunnel Support for IPv4 and IPv6 Packets	411
Configuration Examples for IPv6 GRE Tunnels in CLNS Networks	412
Example: Configuring CTunnels in GRE Mode to Carry IPv6 Packets in CLNS	412
Additional References	412
Feature Information for IPv6 GRE Tunnels in CLNS Networks	413
ISATAP Tunnel Support for IPv6	415
Finding Feature Information	415
Information About ISATAP Tunnel Support for IPv6	415
Overlay Tunnels for IPv6	415
ISATAP Tunnels	418
How to Configure ISATAP Tunnel Support for IPv6	418
Configuring ISATAP Tunnels	418
Configuration Examples for ISATAP Tunnel Support for IPv6	420
Example: Configuring ISATAP Tunnels	420
Additional References	420
Feature Information for ISATAP Tunnel Support for IPv6	421
Manually Configured IPv6 over IPv4 Tunnels	423
Finding Feature Information	423
Information About Manually Configured IPv6 over IPv4 Tunnels	423
Overlay Tunnels for IPv6	423
IPv6 Manually Configured Tunnels	426

How to Enable Manually Configured IPv6 over IPv4 Tunnels	426
Configuring Manual IPv6 Tunnels	426
Configuration Examples for Manually Configured IPv6 over IPv4 Tunnels	427
Example: Configuring Manual IPv6 Tunnels	428
Additional References	428
Feature Information for Manually Configured IPv6 over IPv4 Tunnels	429
IPv6 over IPv4 GRE Tunnels	431
Finding Feature Information	431
Information About IPv6 over IPv4 GRE Tunnels	431
Overlay Tunnels for IPv6	431
GRE IPv4 Tunnel Support for IPv6 Traffic	434
How to Configure IPv6 over IPv4 GRE Tunnels	434
Configuring GRE IPv6 Tunnels	434
Configuration Examples for IPv6 over IPv4 GRE Tunnels	435
Example: GRE Tunnel Running IS-IS and IPv6 Traffic	436
Example: Tunnel Destination Address for IPv6 Tunnel	436
Additional References	437
Feature Information for IPv6 over IPv4 GRE Tunnels	438



IP over IPv6 Tunnels

IPv6 supports IP over IPv6 tunnels, which includes the following:

- Overlay tunneling—Encapsulates IPv6 packets in IPv4 packets for delivery across an IPv4 infrastructure (a core network or the Internet). By using overlay tunnels, you can communicate with isolated IPv6 networks without upgrading the IPv4 infrastructure between them.
- IPv6 manually configured tunnels—A manually configured tunnel is equivalent to a permanent link between two IPv6 domains over an IPv4 backbone. The primary use is for stable connections that require regular secure communication between two edge devices or between an end system and an edge device, or for connection to remote IPv6 networks.
- Generic routing encapsulation (GRE) IPv4 tunnel support for IPv6 traffic—IPv6 traffic can be carried over IPv4 GRE tunnels using the standard GRE tunneling technique that is designed to provide the services to implement any standard point-to-point encapsulation scheme. The primary use of GRE tunnels is for stable connections that require regular secure communication between two edge devices or between an edge device and an end system. The edge devices and the end systems must be dual-stack implementations.
- GRE support over IPv6 transport—GRE has a protocol field that identifies the passenger protocol. GRE tunnels allow Intermediate System-to-Intermediate System (IS-IS) or IPv6 to be specified as a passenger protocol, which allows both IS-IS and IPv6 traffic to run over the same tunnel.
- [Finding Feature Information, page 1](#)
- [Information About IP over IPv6 Tunnels, page 2](#)
- [How to Configure IP over IPv6 Tunnels, page 5](#)
- [Configuration Examples for IP over IPv6 Tunnels, page 8](#)
- [Additional References, page 9](#)
- [Feature Information for IP over IPv6 Tunnels, page 10](#)

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.

Information About IP over IPv6 Tunnels

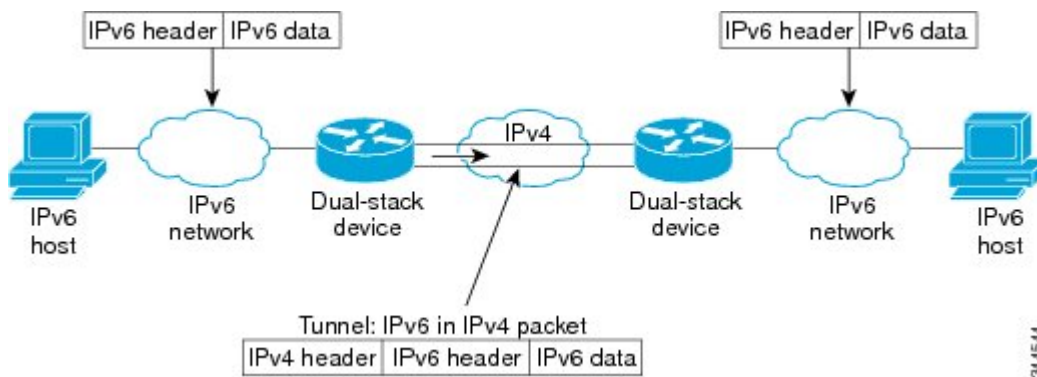
- [Overlay Tunnels for IPv6, page 2](#)
- [IPv6 Manually Configured Tunnels, page 4](#)
- [GRE IPv4 Tunnel Support for IPv6 Traffic, page 4](#)
- [GRE Support over IPv6 Transport, page 4](#)

Overlay Tunnels for IPv6

Overlay tunneling encapsulates IPv6 packets in IPv4 packets for delivery across an IPv4 infrastructure (a core network or the figure below). By using overlay tunnels, you can communicate with isolated IPv6 networks without upgrading the IPv4 infrastructure between them. Overlay tunnels can be configured between border devices or between a border device and a host; however, both tunnel endpoints must support both the IPv4 and IPv6 protocol stacks. IPv6 supports the following types of overlay tunneling mechanisms:

- Manual
- Generic routing encapsulation (GRE)
- IPv4-compatible
- 6to4
- Intrasite Automatic Tunnel Addressing Protocol (ISATAP)

Figure 1 *Overlay Tunnels*



Note

Overlay tunnels reduce the maximum transmission unit (MTU) of an interface by 20 octets (assuming that the basic IPv4 packet header does not contain optional fields). A network that uses overlay tunnels is difficult to troubleshoot. Therefore, overlay tunnels that connect isolated IPv6 networks should not be considered a final IPv6 network architecture. The use of overlay tunnels should be considered as a transition technique toward a network that supports both the IPv4 and IPv6 protocol stacks or just the IPv6 protocol stack.

Use the table below to help you determine which type of tunnel that you want to configure to carry IPv6 packets over an IPv4 network.

Table 1 Suggested Usage of Tunnel Types to Carry IPv6 Packets over an IPv4 Network

Tunneling Type	Suggested Usage	Usage Notes
Manual	Simple point-to-point tunnels that can be used within a site or between sites.	Can carry IPv6 packets only.
GRE-compatible and IPv4-compatible	Simple point-to-point tunnels that can be used within a site or between sites.	Can carry IPv6, Connectionless Network Service (CLNS), and many other types of packets.
IPv4-compatible	Point-to-multipoint tunnels.	Uses the ::96 prefix. We do not recommend using this tunnel type.
6to4	Point-to-multipoint tunnels that can be used to connect isolated IPv6 sites.	Sites use addresses from the 2002::/16 prefix.
6RD	IPv6 service is provided to customers over an IPv4 network by using encapsulation of IPv6 in IPv4.	Prefixes can be from the SP's own address block.
ISATAP	Point-to-multipoint tunnels that can be used to connect systems within a site.	Sites can use any IPv6 unicast addresses.

Individual tunnel types are discussed in detail in this document. We recommend that you review and understand the information about the specific tunnel type that you want to implement. When you are familiar with the type of tunnel you need, see the table below for a summary of the tunnel configuration parameters that you may find useful.

Table 2 Tunnel Configuration Parameters by Tunneling Type

Tunneling Type	Tunnel Configuration Parameter			
Tunnel Mode	Tunnel Source	Tunnel Destination	Interface Prefix or Address	
Manual	ipv6ip	An IPv4 address, or a reference to an interface on which IPv4 is configured.	An IPv4 address.	An IPv6 address.
GRE/IPv4	gre ip		An IPv4 address.	An IPv6 address.
			Not required. These are all point-to-multipoint tunneling types. The IPv4 destination address	Not required. The interface address is generated as :: <i>tunnel-source</i> /96.

Tunneling Type	Tunnel Configuration Parameter		
6to4	ipv6ip 6to4	is calculated, on a per-packet basis, from the IPv6 destination.	An IPv6 address. The prefix must embed the tunnel source IPv4 address.
6RD	ipv6ip 6rd		An IPv6 address.
ISATAP	ipv6ip isatap		An IPv6 prefix in modified eui-64 format. The IPv6 address is generated from the prefix and the tunnel source IPv4 address.

IPv6 Manually Configured Tunnels

A manually configured tunnel is equivalent to a permanent link between two IPv6 domains over an IPv4 backbone. The primary use is for stable connections that require regular secure communication between two edge devices or between an end system and an edge device, or for connection to remote IPv6 networks.

An IPv6 address is manually configured on a tunnel interface, and manually configured IPv4 addresses are assigned to the tunnel source and the tunnel destination. The host or device at each end of a configured tunnel must support both the IPv4 and IPv6 protocol stacks. Manually configured tunnels can be configured between border devices or between a border device and a host. Cisco Express Forwarding switching can be used for IPv6 manually configured tunnels, or Cisco Express Forwarding switching can be disabled if process switching is needed.

GRE IPv4 Tunnel Support for IPv6 Traffic

IPv6 traffic can be carried over IPv4 GRE tunnels using the standard GRE tunneling technique that is designed to provide the services to implement any standard point-to-point encapsulation scheme. As in IPv6 manually configured tunnels, GRE tunnels are links between two points, with a separate tunnel for each link. The tunnels are not tied to a specific passenger or transport protocol but, in this case, carry IPv6 as the passenger protocol with the GRE as the carrier protocol and IPv4 or IPv6 as the transport protocol.

The primary use of GRE tunnels is for stable connections that require regular secure communication between two edge devices or between an edge device and an end system. The edge devices and the end systems must be dual-stack implementations.

GRE Support over IPv6 Transport

GRE has a protocol field that identifies the passenger protocol. GRE tunnels allow Intermediate System-to-Intermediate System (IS-IS) or IPv6 to be specified as a passenger protocol, which allows both IS-IS and IPv6 traffic to run over the same tunnel. If GRE did not have a protocol field, it would be impossible to

distinguish whether the tunnel was carrying IS-IS or IPv6 packets. The GRE protocol field makes it desirable to tunnel IS-IS and IPv6 inside GRE.

How to Configure IP over IPv6 Tunnels

- [Configuring Manual IPv6 Tunnels, page 5](#)
- [Configuring GRE on IPv6 Tunnels, page 7](#)

Configuring Manual IPv6 Tunnels

With manually configured IPv6 tunnels, an IPv6 address is configured on a tunnel interface, and manually configured IPv4 addresses are assigned to the tunnel source and the tunnel destination. The host or device at each end of a configured tunnel must support both the IPv4 and IPv6 protocol stacks.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *tunnel-number*
4. Enter one of the following commands:
 - **ipv6 address** { *ipv6-address/prefix-length* | *prefix-name sub-bits/prefix-length* }
 - **ipv6 address** *ipv6-prefix/prefix-length* [**eui-64**]
5. **tunnel source** { *ip-address* | *interface-type interface-number* }
6. **tunnel destination** *ip-address*
7. **tunnel mode ipv6ip**
8. **end**

DETAILED STEPS

Command or Action	Purpose
Step 1 enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2 configure terminal Example: Device# configure terminal	Enters global configuration mode.

Command or Action	Purpose
<p>Step 3 <code>interface tunnel tunnel-number</code></p> <p>Example:</p> <pre>Device(config)# interface tunnel 0</pre>	<p>Specifies a tunnel interface and number, and enters interface configuration mode.</p>
<p>Step 4 Enter one of the following commands:</p> <ul style="list-style-type: none"> • <code>ipv6 address {ipv6-address/prefix-length prefix-name sub-bits/prefix-length}</code> • <code>ipv6 address ipv6-prefix/prefix-length [eui-64]</code> <p>Example:</p> <pre>Device(config-if)# ipv6 address 3ffe:b00:c18:1::3/127</pre>	<p>Specifies the IPv6 network assigned to the interface and enables IPv6 processing on the interface.</p> <ul style="list-style-type: none"> • If you specify the eui-64 keyword, the software configures an IPv6 address for an interface and enables IPv6 processing on the interface using an EUI-64 interface ID in the low-order 64 bits of the address. <p>Note See the “Implementing IPv6 Addressing and Basic Connectivity” module for more information on configuring IPv6 addresses.</p>
<p>Step 5 <code>tunnel source {ip-address interface-type interface-number}</code></p> <p>Example:</p> <pre>Device(config-if)# tunnel source gigabitethernet 0/0/0</pre>	<p>Specifies the source IPv4 address or the source interface type and number for the tunnel interface.</p> <ul style="list-style-type: none"> • If an interface is specified, the interface must be configured with an IPv4 address.
<p>Step 6 <code>tunnel destination ip-address</code></p> <p>Example:</p> <pre>Device(config-if)# tunnel destination 192.168.30.1</pre>	<p>Specifies the destination IPv4 address or hostname for the tunnel interface.</p>
<p>Step 7 <code>tunnel mode ipv6ip</code></p> <p>Example:</p> <pre>Device(config-if)# tunnel mode ipv6ip</pre>	<p>Specifies a manual IPv6 tunnel.</p> <p>Note The tunnel mode ipv6ip command specifies IPv6 as the passenger protocol and IPv4 as both the encapsulation and transport protocol for the manual IPv6 tunnel.</p>
<p>Step 8 <code>end</code></p> <p>Example:</p> <pre>Device(config-if)# end</pre>	<p>Returns to privileged EXEC mode.</p>

Configuring GRE on IPv6 Tunnels

GRE tunnels can be configured to run over an IPv6 network layer and to transport IPv4 and IPv6 packets in IPv6 tunnels.

When GRE IPv6 tunnels are configured, IPv6 addresses are assigned to the tunnel source and the tunnel destination. The tunnel interface can have either IPv4 addresses or IPv6 addresses assigned (this is not shown in the task). The host or device at each end of a configured tunnel must support both the IPv4 and IPv6 protocol stacks.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel *tunnel-number***
4. Enter one of the following commands:
 - **ipv6 address** {*ipv6-address/prefix-length* | *prefix-name sub-bits/prefix-length*}
 - **ipv6 address** *ipv6-prefix/prefix-length* [**eui-64**]
5. **tunnel source** {*ip-address* | *ipv6-address* | *interface-type interface-number*}
6. **tunnel destination** {*hostname* | *ip-address* | *ipv6-address*}
7. **tunnel mode** {**aurp** | **cayman** | **dvmrp** | **eon** | **gre** | **gre multipoint** | **gre ipv6** | **ipip** [**decapsulate-any**] | **iptalk** | **ipv6** | **mpls** | **nos**}
8. **end**

DETAILED STEPS

Command or Action	Purpose
Step 1 enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2 configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3 interface tunnel <i>tunnel-number</i> Example: Device(config)# interface tunnel 0	Specifies a tunnel interface and number, and enters interface configuration mode.

Command or Action	Purpose
<p>Step 4 Enter one of the following commands:</p> <ul style="list-style-type: none"> • ipv6 address { <i>ipv6-address/prefix-length</i> <i>prefix-name sub-bits/prefix-length</i> } • ipv6 address <i>ipv6-prefix/prefix-length</i> [eui-64] <p>Example:</p> <pre>Device(config-if)# ipv6 address 3ffe:b00:c18:1::3/127</pre>	<p>Specifies the IPv6 network assigned to the interface and enables IPv6 processing on the interface.</p> <ul style="list-style-type: none"> • If you specify the eui-64 keyword, the software configures an IPv6 address for an interface and enables IPv6 processing on the interface using an EUI-64 interface ID in the low-order 64 bits of the address.
<p>Step 5 tunnel source { <i>ip-address</i> <i>ipv6-address</i> <i>interface-type interface-number</i> }</p> <p>Example:</p> <pre>Device(config-if)# tunnel source gigabitethernet 0/0/0</pre>	<p>Specifies the source IPv4 address, IPv6 address, or the source interface type and number for the tunnel interface.</p> <ul style="list-style-type: none"> • If an interface is specified, the interface must be configured with an IPv4 address.
<p>Step 6 tunnel destination { <i>hostname</i> <i>ip-address</i> <i>ipv6-address</i> }</p> <p>Example:</p> <pre>Device(config-if)# tunnel destination 2001:DB8:1111:2222::1/64</pre>	<p>Specifies the destination IPv4 address, IPv6 address, or hostname for the tunnel interface.</p>
<p>Step 7 tunnel mode { <i>aurp</i> <i>cayman</i> <i>dvmrp</i> <i>eon</i> gre gre multipoint gre ipv6 ipip [<i>decapsulate-any</i>] iptalk ipv6 <i>mpls</i> <i>nos</i> }</p> <p>Example:</p> <pre>Device(config-if)# tunnel mode gre ipv6</pre>	<p>Specifies a GRE IPv6 tunnel.</p> <p>Note The tunnel mode gre ipv6 command specifies GRE as the encapsulation protocol for the tunnel.</p>
<p>Step 8 end</p> <p>Example:</p> <pre>Device(config-if)# end</pre>	<p>Returns to privileged EXEC mode.</p>

Configuration Examples for IP over IPv6 Tunnels

- [Example: Configuring Manual IPv6 Tunnels, page 9](#)
- [Example: Configuring GRE on IPv6 Tunnels, page 9](#)

Example: Configuring Manual IPv6 Tunnels

The following example shows how to manually configure an IPv6 tunnel between Device A and Device B. In the example, tunnel interface 0 for both Device A and Device B is manually configured with a global IPv6 address. The tunnel source and destination addresses are also manually configured.

Device A

```
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# ip address 192.168.99.1 255.255.255.0
Device(config-if)# exit
Device(config-if)# interface tunnel 0
Device(config-if)# ipv6 address 2001:0db8:c18:1::3/126
Device(config-if)# tunnel source GigabitEthernet 0/0/0
Device(config-if)# tunnel destination 192.168.30.1
Device(config-if)# tunnel mode ipv6ip
Device(config-if)# end
```

Device B

```
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# ip address 192.168.30.1 255.255.255.0
Device(config-if)# exit
Device(config-if)# interface tunnel 0
Device(config-if)# ipv6 address 2001:0db8:c18:1::2/126
Device(config-if)# tunnel source GigabitEthernet 0/0/0
Device(config-if)# tunnel destination 192.168.99.1
Device(config-if)# tunnel mode ipv6ip
Device(config-if)# end
```

Example: Configuring GRE on IPv6 Tunnels

The following example shows how to configure a GRE tunnel over an IPv6 transport. Gigabit Ethernet interface 0/0/0 is configured with an IPv6 address, and this is the source address used by the tunnel interface. The destination IPv6 address of the tunnel is specified directly. In the following example, the tunnel carries both IPv4 and IS-IS traffic:

```
Device(config)# interface tunnel 0
Device(config-if)# ip address 10.1.1.1 255.255.255.0
Device(config-if)# ip router isis
Device(config-if)# tunnel source GigabitEthernet 0/0/0
Device(config-if)# tunnel destination 2001:DB8:1111:2222::1
Device(config-if)# tunnel mode gre ipv6
Device(config-if)# exit
Device(config)# interface FastEthernet 0/0
Device(config-if)# no ip address
Device(config-if)# ipv6 address 2001:DB8:1111:1111::1/64
Device(config-if)# exit
Device(config)# router isis
Device(config-router)# net 49.0001.0000.0000.000a.00
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	<i>Cisco IOS Master Command List, All Releases</i>
IPv6 commands	<i>Cisco IOS IPv6 Command Reference</i>
IPv6 addressing and connectivity	<i>IPv6 Configuration Guide</i>
Cisco IOS IPv6 features	<i>Cisco IOS IPv6 Feature Mapping</i>

Standards and RFCs

Standard/RFC	Title
RFCs for IPv6	<i>IPv6 RFCs</i>

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for IP over IPv6 Tunnels

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 3 **Feature Information for IP over IPv6 Tunnels**

Feature Name	Releases	Feature Information
IP over IPv6 Tunnels	12.2(30)S 12.2(33)SRA 12.3(7)T 12.4 12.4(2)T 15.0(1)S Cisco IOS XE Release 2.1 15.1(1)SY	IP over IPv6 Tunnels feature is supported. The following commands were introduced or modified: tunnel destination , tunnel mode , tunnel mode ipv6ip , tunnel source .

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



Configuring LAN Interfaces

Use the information in this chapter to configure LAN interfaces supported on Cisco routers and access servers.

To identify the hardware platform or software image information associated with a feature, use Cisco Feature Navigator on Cisco.com to search for information about the feature.

This chapter describes the processes for configuring LAN interfaces and includes the following sections:

For examples of configuration tasks, see the [LAN Interface Configuration Examples](#), page 51.

For hardware technical descriptions and information about installing interfaces, refer to the hardware installation and configuration publication for your product. For a complete description of the LAN interface commands used in this chapter, refer to the Cisco IOS Interface and Hardware Component Command Reference . To locate documentation of other commands that appear in this chapter, use the master commands list or search online.

- [Finding Feature Information](#), page 13
- [Configuring Ethernet Fast Ethernet or Gigabit Ethernet Interfaces](#), page 14
- [Configuring the Cisco 7200-I O-GE+E and Cisco 7200-I O-2FE E Input Output Controllers](#), page 27
- [Configuring Fast EtherChannel](#), page 32
- [Configuring a FDDI Interface](#), page 38
- [Configuring a Hub Interface](#), page 46
- [Configuring a Token Ring Interface](#), page 48
- [LAN Interface Configuration Examples](#), page 51

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.

Configuring Ethernet Fast Ethernet or Gigabit Ethernet Interfaces

Cisco supports 10-Mbps Ethernet, 100-Mbps Fast Ethernet, and 1000-Mbps Gigabit Ethernet. Support for the 10-Mbps, 100-Mbps, and 1000-Mbps Ethernet interface is supplied on various Ethernet network interface cards or systems.

Fast Ethernet NP-1FE Module Benefits

- VLAN routing--VLAN support enables network managers to group users logically rather than by physical location. The high performance of the underlying Cisco 4700, combined with the feature-rich NP-1FE, makes it an ideal combination for a low-density, higher-performance application such as inter-VLAN routing.
- High-speed interconnections--The Fast Ethernet interface enables network managers to implement Fast-Ethernet routing solutions for optimal cost and performance across a wide range of applications, including campus or enterprise backbones and data centers. It is also a low-cost way to provide Fast-Ethernet access to traditional low-speed WAN services.
- Local area network aggregation--The Cisco 4500 or the Cisco 4700 series routers can support as many as 12 Ethernet, 4 Token Ring, or 1 FDDI segment. ISDN interfaces are also supported.

With the Catalyst 3000 or Catalyst 5000 system, the Fast Ethernet processor can be used to aggregate up to twelve 10-Mbps LANs and give them high-speed access to such Layer 3 routing services as providing firewalls and maintaining access lists.

Cisco 7200 Series Routers with Fast Ethernet and Gigabit Ethernet

Cisco 7200 series routers support an I/O controller with an RJ-45 interface for Fast Ethernet support and an I/O controller with both RJ-45 and GBIC interfaces for Gigabit Ethernet support.

The Cisco 7200-I/O-GE+E is an Input/Output controller that provides one Gigabit Ethernet and one Ethernet port. It is equipped with a GBIC receptacle for 1000 Mbps operation and an RJ-45 receptacle for 10-Mbps operation.

The Cisco 7200-I/O-2FE/E is an I/O controller that provides two autosensing Fast Ethernet ports and is equipped with two RJ-45 receptacles for 10/100 Mbps operation.

You can configure the Fast Ethernet port for use at 100-Mbps full-duplex or half-duplex operation (half duplex is the default). The Fast Ethernet port is equipped with either a single MII receptacle or an MII receptacle and an RJ-45 receptacle. To support this new feature, the **media-type** interface command has been modified. The **media-type** command now supports two options:

- **100BASE-X** --Specifies an RJ-45 100BASE-X physical connection.
- **mii** --Specifies a media-independent interface.

The Gigabit Ethernet interface on the Cisco 7200-I/O-GE+E operates at full duplex and cannot be configured for half-duplex mode.

Second-generation Fast Ethernet Interface Processors (FEIP2-DSW-2TX and FEIP2-DSW-2FX) are available on Cisco 7500 series routers and on Cisco 7000 series routers with the 7000 Series Route Switch Processor (RSP7000) and 7000 Series Chassis Interface (RSP7000CI). The FEIP2-DSW is a dual-port, fixed-configuration interface processor that provides two 100-Mbps Fast Ethernet (FE) interfaces. Each interface on the FEIP2-DSW supports both half-duplex and full-duplex.

Refer to the *Cisco Product Catalog* for specific platform and hardware compatibility information.

Use the **show interfaces**, **show controllers mci**, and **show controllers cbusEXEC** commands to display the Ethernet port numbers. These commands provide a report for each interface supported by the router or access server.

Use the **show interfaces fastethernet** command to display interface statistics, and use the **show controllers fastethernet** to display information about the Fast Ethernet controller chip. The output shows statistics, including information about initialization block information, transmit ring, receive ring, and errors.

Use the **show interfaces gigabitethernet** command to display interface statistics, and use the **show controllers gigabitethernet** to display the information about the Gigabit Ethernet controller chip. The output shows statistics, including information about initialization block information, transmit ring, receive ring, and errors.

For information on how to configure Fast EtherChannel, see the tasks listed in the [Configuring Fast EtherChannel](#), page 32.

- [Ethernet Fast Ethernet and Gigabit Ethernet Interface Configuration Task List](#), page 15

Ethernet Fast Ethernet and Gigabit Ethernet Interface Configuration Task List

To configure features on an Ethernet, Fast Ethernet, or Gigabit Ethernet interface, perform the tasks in the following sections:

- [Specifying an Ethernet Fast Ethernet or Gigabit Ethernet Interface](#), page 15
- [Specifying an Ethernet Encapsulation Method](#), page 16
- [Specifying Full-Duplex Operation](#), page 16
- [Specifying the Media and Connector Type](#), page 17
- [Extending the 10BASE-T Capability](#), page 18
- [Configuring Fast Ethernet 100BASE-T](#), page 18
- [Configuring PA-12E 2FE Port Adapters](#), page 20
- [Configuring the 100VG-AnyLAN Port Adapter](#), page 26

Specifying an Ethernet Fast Ethernet or Gigabit Ethernet Interface

To specify an Ethernet interface and enter interface configuration mode, use one of the following commands in global configuration mode.

Command or Action	Purpose
Router(config)# interface ethernet <i>number</i>	Enters interface configuration mode.
Router(config)# interface ethernet <i>slot / port</i>	Enters interface configuration mode for the Cisco 7200 and Cisco 7500 series routers.
Router(config)# interface ethernet <i>slot / port-adapter / port</i>	Enters interface configuration mode for Cisco 7500 series routers.
Router(config)# interface fastethernet <i>number</i>	Enters interface configuration mode for the Cisco 4000 series with a Fast Ethernet NIM installed.

Command or Action	Purpose
Router(config)# interface fastethernet slot / port	Specifies a Fast Ethernet interface and enters interface configuration mode on the Cisco 7200 series routers.
Router(config)# interface fastethernet slot / port-adapter / port	Specifies a Fast Ethernet interface and enters interface configuration mode on the Cisco 7500 series routers.
Router(config)# interface gigabitethernet slot / port	Specifies a Gigabit Ethernet interface and enters interface configuration mode on the Cisco 7200 series routers.

To display the Fast Ethernet slots and ports, use the **show interfaces fastethernet** command. The Fast Ethernet network interface module (NIM) and the Fast Ethernet Interface Processor (FEIP) default to half-duplex mode.

Specifying an Ethernet Encapsulation Method

Currently, there are three common Ethernet encapsulation methods:

- The standard Advanced Research Projects Agency (ARPA) Ethernet Version 2.0 encapsulation, which uses a 16-bit protocol type code (the default encapsulation method).
- Service access point (SAP) IEEE 802.3 encapsulation, in which the type code becomes the frame length for the IEEE 802.2 LLC encapsulation (destination and source Service Access Points, and a control byte).
- The SNAP method, as specified in RFC 1042, *>Standard for the Transmission of IP Datagrams Over IEEE 802 Networks*, which allows Ethernet protocols to run on IEEE 802.2 media.

The encapsulation method that you use depends upon the routing protocol that you are using, the type of Ethernet media connected to the router or access server, and the routing or bridging application that you configure.

To establish Ethernet encapsulation of IP packets, use one of the following commands in interface configuration mode.

Command or Action	Purpose
Router(config-if)# encapsulation arpa	Selects ARPA Ethernet encapsulation.
Router(config-if)# encapsulation sap	Selects SAP Ethernet encapsulation.
Router(config-if)# encapsulation snap	Selects SNAP Ethernet encapsulation.

For an example of selecting Ethernet encapsulation for IP, see the [Ethernet Encapsulation Enablement Example, page 51](#).

Specifying Full-Duplex Operation

The default is half-duplex mode on the FEIP2-DSW-2FX. To enable full-duplex mode on the FEIP2-DSW-2FX (for a maximum aggregate bandwidth of 200 Mbps), use either of the following commands in interface configuration mode.

Command or Action	Purpose
Router(config-if)# full-duplex	Enables full-duplex on the Fast Ethernet interface of the FEIP2-DSW-2FX.
or	
Router(config-if)# no half-duplex	

For an example of enabling full-duplex mode on Fast Ethernet, see the [Full-Duplex Enablement Operation Example](#), page 52.



Caution

To prevent system problems, do not configure both FEIP2-DSW-2FX interfaces for full-duplex operation at the same time. The FEIP2-DSW-2TX supports half-duplex only and should not be configured for full-duplex.

Specifying the Media and Connector Type

You can specify that the Ethernet network interface module (NIM) on the Cisco 4000 series routers use either the default of an attachment unit interface (AUI) and a 15-pin connector, or 10BASE-T and an RJ-45 connector. To do so, use one of the following commands in interface configuration mode.

Command or Action	Purpose
Router(config-if)# media-type aui	Selects a 15-pin Ethernet connector.
Router(config-if)# media-type 10baset	Selects an RJ-45 Ethernet connector.

The default media connector type is an RJ-45 or SC (fiber-optic) connector. You can specify that the interface uses either an MII connector, or an RJ-45 or SC (fiber-optic) connector (this is the default). To do so, use one of the following commands in interface configuration mode.

Command or Action	Purpose
Router(config-if)# media-type mii	Selects an MII Ethernet connector.
Router(config-if)# media-type 100baseX	Selects an RJ-45 Ethernet connector for the FEIP2-DSW-2TX or an SC connector for the FEIP2-DSW-2FX.

**Note**

When using the I/O controller that is equipped with an MII receptacle and an RJ-45 receptacle, only one receptacle can be configured for use at a time.

Extending the 10BASE-T Capability

On a Cisco 4000 series or Cisco 4500 series routers, you can extend the twisted-pair 10BASE-T capability beyond the standard 100 meters by reducing the *squelch* (signal cutoff time). This feature applies only to the LANCE controller 10BASE-T interfaces. LANCE is the AMD controller chip for the Cisco 4000 and Cisco 4500 Ethernet interface and does not apply to the Fast Ethernet interface.

To reduce squelch, use the first command in the following table in interface configuration mode. You can later restore the squelch by using the second command.

Command or Action	Purpose
Router(config-if)# squelch reduced	Reduces the squelch.
Router(config-if)# squelch normal	Returns squelch to normal.

Configuring Fast Ethernet 100BASE-T

You must configure the Fast Ethernet 100BASE-T interface on a Cisco AS5300 so that it can be recognized as a device on the Ethernet LAN. The Fast Ethernet interface supports 10- and 100-Mbps speeds with the 10BASE-T and 100BASE-T routers, hubs, and switches.

To configure the interface, use the following commands beginning in privileged EXEC mode.

SUMMARY STEPS

1. Router# **configure terminal**
2. Router(config)# **interface fastethernet** *number*
3. Router(config-if)# **ip address** *address subnet-mask*
4. Router(config-if)# **speed** { **10** | **100** | **auto** }
5. Router(config-if)# **duplex** { **full** | **half** | **auto** }

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# interface fastethernet <i>number</i>	Enters Fast Ethernet interface configuration mode.
Step 3	Router(config-if)# ip address <i>address subnet-mask</i>	Assigns an IP address and subnet mask to the interface.

Command or Action	Purpose
Step 4 Router(config-if)# speed { 10 100 auto }	Assigns a speed to the interface. The default is 100 Mbps. ¹ For relationship between duplex and speed command options, see the table below.
Step 5 Router(config-if)# duplex { full half auto }	Sets up the duplex configuration on the Fast Ethernet interface. The default is half duplex. Configuring Fast Ethernet 100BASE-T, page 18 For relationship between duplex and speed command options, see the table below.

To use the autonegotiation capability (that is, to detect speed and duplex modes automatically), you must set both **speed** and **duplex** command to **auto**. Setting the **speed** command to **auto** negotiates speed only, and setting **duplex** command to **auto** negotiates duplex only. The table below describes the performance of the access server for different combinations of the **duplex** and **speed** command options. The specified **duplex** command option plus the specified **speed** command option produces the resulting system action.

Table 4 Relationship Between duplex and speed Command Options

duplex Command	speed Command	Resulting System Actions
Router(config-if)# duplex auto	auto	speed Autonegotiates both speed and duplex modes.
Router(config-if)# duplex auto	10 or speed 100	speed Autonegotiates both speed and duplex modes.
Router(config-if)# duplex half	auto	speed Autonegotiates both speed and duplex modes.
or Router(config-if)# duplex full		
Router(config-if)# duplex half	speed 10	Forces 10 Mbps and half duplex.
Router(config-if)# duplex full	speed 10	Forces 10 Mbps and full duplex.

¹ The auto option automatically negotiates the speed on the basis of the speed and the peer router, hub, or switch media.

duplex Command	speed Command	Resulting System Actions
Router(config-if)# duplex half	100	Forces 100 Mbps and half duplex.
Router(config-if)# duplex full	100	Forces 100 Mbps and full duplex.

Configuring PA-12E 2FE Port Adapters

The PA-12E/2FE Ethernet switch port adapter provides Cisco 7200 series routers with up to twelve 10-Mbps and two 10/100-Mbps switched Ethernet (10BASE-T) and Fast Ethernet (100BASE-TX) interfaces for an aggregate bandwidth of 435 Mbps, full-duplex. The PA-12E/2FE port adapter supports the Ethernet, IEEE 802.3, and IEEE 802.3u specifications for 10-Mbps and 100-Mbps transmission over unshielded twisted pair (UTP) cables.

The PA-12E/2FE port adapter offloads Layer 2 switching from the host CPU by using store-and-forward or cut-through switching technology between interfaces within the same virtual LAN (VLAN) on the PA-12E/2FE port adapter. The PA-12E/2FE port adapter supports up to four VLANs (bridge groups).



Note

The PA-12E/2FE port adapter is a dual-width port adapter, which means it occupies two horizontally aligned port adapter slots when installed in a Cisco 7200 series router. (Single-width port adapters occupy individual port adapter slots in a Cisco 7200 series router.)

All interfaces on the PA-12E/2FE port adapter support autosensing and autonegotiation of the proper transmission mode (half-duplex or full-duplex) with an attached device. The first two PA-12E/2FE interfaces (port 0 and port 1) also support autosensing and autonegotiation of the proper connection speed (10 Mbps or 100 Mbps) with an attached device. If an attached device does not support autosensing and autonegotiation of the proper transmission mode, the PA-12E/2FE interfaces attached to the device automatically enter half-duplex mode. Use the **show system:running-config** command to determine if a PA-12E/2FE interface is autosensing and autonegotiating the proper transmission mode with an attached device. Use the **full-duplex** and the **half-duplex** commands to change the transmission mode of a PA-12E/2FE interface. After changing the transmission mode, use the **show interfaces** command to verify the transmission mode of the interface.



Note

If you use the **full-duplex** and the **half-duplex** commands to change the transmission mode of the first two PA-12E/2FE interfaces (port 0 and port 1), the transmission speed of the two PA-12E/2FE interfaces automatically defaults to 100-Mbps. The first two PA-12E/2FE interfaces operate only at 10 Mbps when the interfaces are autosensing and autonegotiating the proper connection speed (10 Mbps or 100 Mbps) with an attached device.

To configure the PA-12E/2FE port adapter, perform the tasks in the following sections. Each task in the list is identified as either required or optional.

**Note**

If you plan to use a PA-12E/2FE interface to boot from a network (using TFTP), ensure that the interface is configured for a loop-free environment, that an IP address is configured for the interface's bridge-group virtual interface, and that system boot image 11.2(10)P is installed on your router (use the **show version** command to view the system boot image of your router). Then, *before* booting from the network server, use the **bridge-group bridge-group number spanning-disabled** command to disable the Spanning Tree Protocol configured on the interface to keep the TFTP server from timing out and closing the session. For detailed information about booting from a network using TFTP, loading a system image from a network server, and configuring the Spanning Tree Protocol on your Cisco 7200 series router, refer to the *PA-12E/2FE Ethernet Switch Port Adapter* book that accompanies the hardware and to the Cisco IOS Bridging and IBM Networking Configuration Guide.

For information on other commands that can be used to configure a PA-12E/2FE port adapter, refer to the *CiscoIOS Interface and Hardware Component Command Reference*. For information on bridging, refer to the "Configuring Transparent Bridging" chapter in the *CiscoIOS Bridging and IBM Networking Configuration Guide*.

For PA-12E/2FE port adapter configuration examples, see the [PA-12E 2FE Port Configuration Examples, page 52](#).

- [Configuring the PA-12E 2FE Port Adapter, page 21](#)
- [Configuring the PA-12E 2FE Port Adapter, page 22](#)
- [Monitoring and Maintaining the PA-12E 2FE Port Adapter, page 23](#)
- [Configuring Bridge Groups Using the 12E 2FE VLAN Configuration WebTool, page 25](#)

Configuring the PA-12E 2FE Port Adapter

This section provides instructions for a basic configuration. You might also need to enter other configuration commands depending on the requirements for your system configuration and the protocols that you plan to route on the interface. For complete descriptions of configuration commands and the configuration options available, refer to the other configuration guides and command references in the Cisco IOS documentation set.

To configure the interfaces on the PA-12E/2FE port adapter, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. Router(config)# **bridge bridge-group protocol ieee**
2. Router(config)# **interface fastethernet slot / port**
3. Router(config-if)# **bridge-group bridge-group**
4. Router(config-if)# **cut-through [receive | transmit]**
5. Router(config-if)# **full-duplex**
6. Router(config-if)# **no shutdown**
7. Router(config-if)# **exit**
8. Repeat Steps 1 through 7 for each interface.
9. Router# **copy system:running-config nvram:startup-config**

DETAILED STEPS

Command or Action	Purpose
Step 1 Router(config)# bridge bridge-group protocol ieee	Specifies the type of Spanning Tree Protocol. The PA-12E/2FE port adapter supports DEC and IEEE Spanning Tree Protocols; however, we recommend using the IEEE protocol when configuring bridge groups.
Step 2 Router(config)# interface fastethernet slot / port Example: (ports 0 and 1) Example: Router(config)# interface ethernet slot / port Example: (ports 2 through 13)	Enters interface configuration mode for the interface that you want to configure.
Step 3 Router(config-if)# bridge-group bridge-group	Assigns a bridge group to the interface.
Step 4 Router(config-if)# cut-through [receive transmit]	(Optional) Configures the interface for cut-through switching technology. The default is store-and-forward (that is, no cut-through).
Step 5 Router(config-if)# full-duplex	(Optional) Configures the transmission mode for full-duplex, if an attached device does not support autosensing or autonegotiation. The default is half-duplex.
Step 6 Router(config-if)# no shutdown	Restarts the interface.
Step 7 Router(config-if)# exit	Returns to global configuration mode.
Step 8 Repeat Steps 1 through 7 for each interface.	--
Step 9 Router# copy system:running-config nvram:startup-config	Saves the new configuration to memory.

Configuring the PA-12E 2FE Port Adapter

To enable integrated routing and bridging on the bridge groups, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. Router(config)# **bridge irb**
2. Router(config)# **interface bvi** *bridge-group*
3. Router(config-if)# **ip address** *ip-address mask*
4. Router(config-if)# **no shutdown**
5. Router(config-if)# **exit**
6. Repeat Steps 1 through 5 for each bridge group.
7. Router(config)# **bridge** *bridge-group* **route protocol**
8. Router(config)# **exit**
9. Router# **copy system:running-config nvram:startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config)# bridge irb	Enables integrated routing and bridging.
Step 2	Router(config)# interface bvi <i>bridge-group</i>	Enables a virtual interface on a bridge group.
Step 3	Router(config-if)# ip address <i>ip-address mask</i>	Assigns an IP address and subnet mask to the bridge-group virtual interface.
Step 4	Router(config-if)# no shutdown	Restarts the interface.
Step 5	Router(config-if)# exit	Returns to global configuration mode.
Step 6	Repeat Steps 1 through 5 for each bridge group.	--
Step 7	Router(config)# bridge <i>bridge-group</i> route protocol	Specifies the protocol for each bridge group.
Step 8	Router(config)# exit	Exits global configuration mode.
Step 9	Router# copy system:running-config nvram:startup-config	Saves the new configuration to memory.

Monitoring and Maintaining the PA-12E 2FE Port Adapter

After configuring the new interface, you can display its status and verify other information. To display information about the PA-12E/2FE port adapter, use the following commands in EXEC mode.

Command or Action	Purpose
Router# show version	Displays the configuration of the system hardware, the software version, the names and sources of configuration files, and the boot image.
Router# show controllers	Displays all current port adapters and their interfaces

Command or Action	Purpose
Router# show interfaces fastethernet <i>slot / port</i> (ports 0 and 1) or Router# show interfaces ethernet <i>slot / port</i> (ports 2 through 13)	Displays the interfaces so that you can verify that they have the correct slot number and that the interface and line protocol are in the correct state.
Router# show bridge group	Displays all bridge groups and their interfaces.
Router# show interfaces fastethernet <i>slot / port</i> irb (ports 0 and 1) or Router# show interfaces ethernet <i>slot / port</i> irb (ports 2 through 13)	Displays the routed protocol so you can verify that it is configured correctly for each interface.
Router# show protocols	Displays the protocols configured for the entire system and specific interfaces.
Router# show pas eswitch addresses fastethernet <i>slot / port</i> (ports 0 and 1) or Router# show pas eswitch addresses ethernet <i>slot / port</i> (ports 2 through 13)	Displays the Layer 2 learned addresses for each interface.
Router# more system:running-config	Displays the running configuration file.
Router# more nvram:startup-config	Displays the configuration stored in NVRAM.

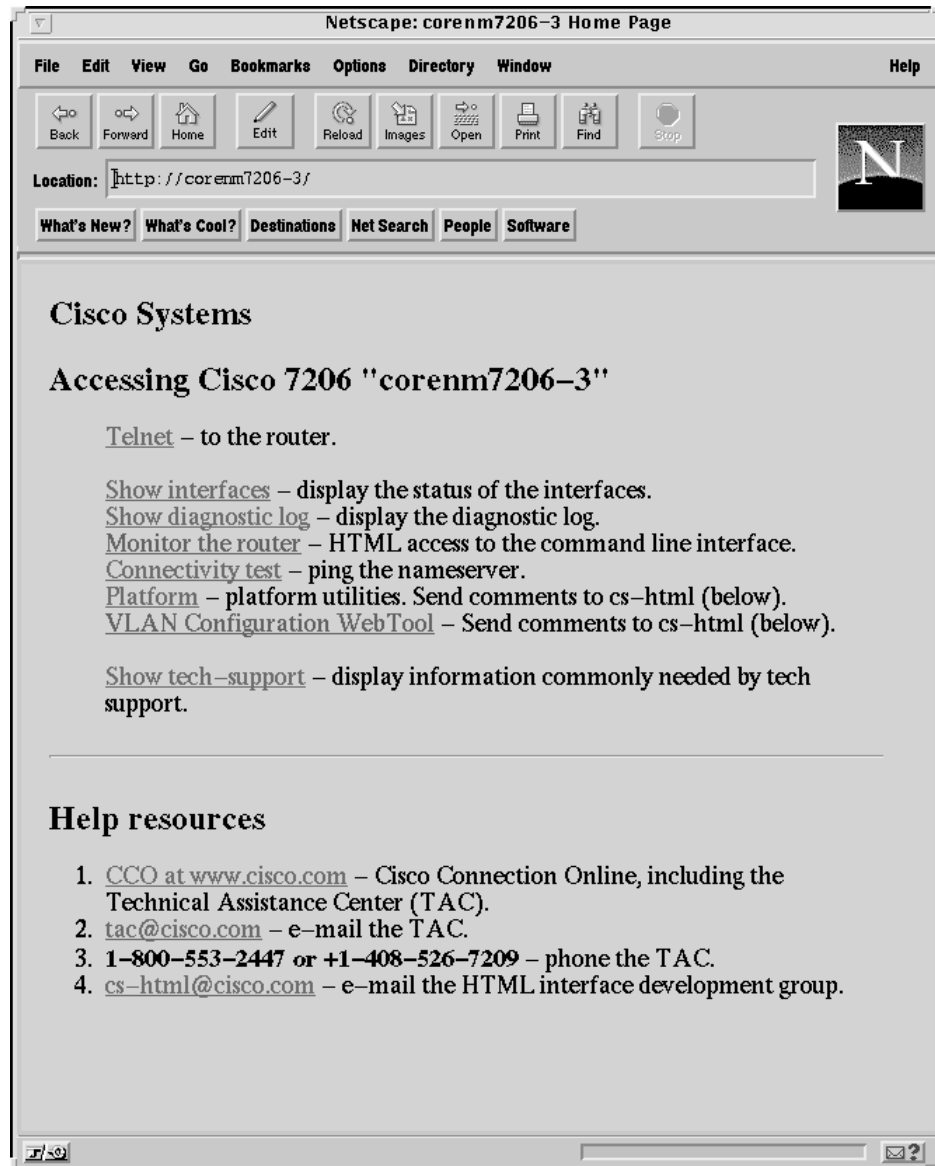
Configuring Bridge Groups Using the 12E 2FE VLAN Configuration WebTool

The 12E/2FE VLAN Configuration WebTool, shown in the figure below, is a web browser-based Java applet that displays configured interfaces and bridge groups for PA-12E/2FE port adapters installed in Cisco routers. With the WebTool you can perform the following tasks:

- Create and delete bridge groups (also referred to as VLANs)
- Add and remove PA-12E/2FE interfaces from bridge groups
- Assign colors to bridge groups and PA-12E/2FE interfaces
- Administratively shut down (disable) and bring up (enable) PA-12E/2FE interfaces
- View the bridge-group status of each PA-12E/2FE interface

You can access the 12E/2FE VLAN Configuration WebTool from the home page of your router. For complete procedures on how to use the VLAN Configuration WebTool, refer to the *PA-12E/2FE Ethernet Switch Port Adapter* book that accompanies the hardware.

Figure 2 Example Home Page for a Cisco 7200 Series Router (Cisco 7206 Shown)



All Cisco routers that run Cisco IOS Release 11.0 or later have a home page. All Cisco router home pages are password protected. Contact your network administrator if you do not have the name or password for your Cisco 7200 series router.

If your router has an installed PA-12E/2FE port adapter, the 12E/2FE VLAN Configuration WebTool shown in the figure above can be accessed from the home page of the router using a Java-enabled web browser.

Configuring the 100VG-AnyLAN Port Adapter

The 100VG-AnyLAN port adapter (PA-100VG) is available on Cisco 7200 series routers and on Cisco 7500 series routers.

The PA-100VG provides a single interface compatible with and specified by IEEE 802.12 to support 100 Mbps over Category 3 or Category 5 UTP cable with RJ-45 terminators. The PA-100VG supports 802.3 Ethernet packets and can be monitored with the IEEE 802.12 Interface MIB.

To configure the PA-100VG port adapter, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. Do one of the following:
 - Router(config)# **interface vg-anylan** *slot / port*
 - (Cisco 7200)
 -
 -
 -
2. Router(config-if)# **ip address** *ip-address mask*
3. Router(config-if)# **frame-type ethernet**

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 Do one of the following:</p> <ul style="list-style-type: none"> • Router(config)# interface vg-anylan <i>slot / port</i> • (Cisco 7200) • • • <p>Example:</p> <pre>Router(config)# interface vg-anylan <i>slot / port-adapter / port</i></pre> <p>Example:</p> <pre>(Cisco 7500)</pre>	<p>Specifies a 100VG-AnyLAN interface and enters interface configuration.</p>

Command or Action	Purpose
Step 2 Router(config-if)# ip address <i>ip-address mask</i>	Specifies the IP address and subnet mask to the interface.
Step 3 Router(config-if)# frame-type ethernet	Configures the frame type. Currently, only Ethernet frames are supported. The frame type defaults to Ethernet.

**Note**

The port number for the 100VG-AnyLAN port adapter is always 0.

Configuring the PA-100VG interface is similar to configuring an Ethernet or Fast Ethernet interface. To display information about the 100VG-AnyLAN port adapter, use the **show interfaces vg-anylan EXEC** command.

Configuring the Cisco 7200-I O-GE+E and Cisco 7200-I O-2FE E Input Output Controllers

The Cisco 7200-I/O-GE+E is an Input/Output controller that provides one Gigabit Ethernet and one Ethernet port. It is equipped with a GBIC receptacle for 1000-Mbps operation and an RJ-45 receptacle for 10-Mbps operation.

The Cisco 7200-I/O-2FE/E is an Input/Output controller that provides two autosensing Fast Ethernet ports and is equipped with two RJ-45 receptacles for 10/100-Mbps operation.

I/O controllers support the following features:

- Dual EIA/TIA-232 channels for local console and auxiliary ports
 - NVRAM for storing the system configuration and environmental monitoring logs
 - Two PC Card slots that hold Flash disks or Flash memory cards for storing the default Cisco IOS software image
 - Flash memory for storing the boot helper image
 - Two environmental sensors for monitoring the cooling air as it enters and leaves the chassis
- [Cisco 7200-I O-GE+E and Cisco 7200-I O-2FE E Configuration Task List, page 27](#)

Cisco 7200-I O-GE+E and Cisco 7200-I O-2FE E Configuration Task List

See the following sections for configuration tasks for the Cisco 7200-I/O-GE+E and the Cisco 7200-I/O-2FE/E feature. Each task in the list is identified as required or optional.

**Note**

For Cisco 7200 VXR routers used as router shelves in AS5800 Universal Access Servers, use the *router-shelf / slot / port* command format for all interface commands.

- [Configuring the Interface Transmission Mode, page 28](#)
- [Configuring Interface Speed, page 28](#)

- [Configuring the Ethernet Fast Ethernet and Gigabit Ethernet Interfaces](#), page 29
- [Verifying the Configuration](#), page 30
- [Monitoring and Maintaining the Cisco 7200-I O GE+E and Cisco 7200-I O-2FE E](#), page 31

Configuring the Interface Transmission Mode

To configure the interface transmission mode, use the following commands beginning in privileged EXEC mode. The Fast Ethernet and Ethernet interfaces on the Cisco 7200-I/O-2FE/E are **duplex auto** by default.

SUMMARY STEPS

1. Router# **configure terminal**
2. Router(config)# **interface fastethernet slot / port²**
3. Router(config)# **duplex full**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# configure terminal	Enters global configuration mode and specifies that the console terminal is the source of the configuration subcommands.
Step 2	Router(config)# interface fastethernet slot / port²	Selects the Fast Ethernet interface to configure.
Step 3	Router(config)# duplex full	Changes the Fast Ethernet interface port transmission mode to full duplex from autonegotiation.

Configuring Interface Speed

To configure the two autosensing Ethernet/Fast Ethernet interfaces on the C7200-I/O-2FE/E, use the **speed** command. The the default interface speed is **auto**.The following procedure configures the C7200-I/O-2FE/E for a speed of 10 Mbps.

SUMMARY STEPS

1. Router# **configure terminal**
2. Router(Config)# **interface ethernet slot / port³**
3. Router(Config-if)# **interface fastethernet slot / port⁴**
4. Router(Config-if)# **speed 10**

² Use the interface fastethernet router-shelf / slot / port command for a Cisco 7200 VXR used as a router shelf in an AS5800 Universal Access Server.

² Use the interface fastethernet router-shelf / slot / port command for a Cisco 7200 VXR used as a router shelf in an AS5800 Universal Access Server.

³ Use the interface ethernet router-shelf / slot / port command for a Cisco 7200 VXR used as a router shelf in an AS5800 Universal Access Server.

⁴ Use the interface fastethernet router-shelf / slot / port command for a Cisco 7200 VXR used as a router shelf in an AS5800 Universal Access Server.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# configure terminal	Enters global configuration mode and specifies that the console terminal is the source of the configuration subcommands.
Step 2	Router(Config)# interface ethernet slot / port ³	Selects the Ethernet interface to configure.
Step 3	Router(Config-if)# interface fastethernet slot / port ⁴	Selects the Fast Ethernet interface to configure.
Step 4	Router(Config-if)# speed 10	Sets the Ethernet or Fast Ethernet interface speed to 10 Mbps.

Configuring the Ethernet Fast Ethernet and Gigabit Ethernet Interfaces

The following procedure explains a basic configuration for an Ethernet, Fast Ethernet, or Gigabit Ethernet interface on a C7200-I/O-GE+E or a C7200-I/O-2FE/E.

SUMMARY STEPS

1. Router# **configure terminal**
2. Router(config)# **interface ethernet slot / port**⁵
3. Router(config)# **interface fastethernet slot / port**⁶
4. Router(config)# **interface gigabitethernet slot / port**⁷
5. Router(config-if) # **ip address ip-address mask**
6. Router(config-if)# **duplex auto**
7. Router#(config-if)# **Exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# configure terminal	Enters global configuration mode and specifies that the console terminal is the source of the configuration subcommands.
	Example:	

³ Use the interface ethernet router-shelf / slot / port command for a Cisco 7200 VXR used as a router shelf in an AS5800 Universal Access Server.

⁴ Use the interface fastethernet router-shelf / slot / port command for a Cisco 7200 VXR used as a router shelf in an AS5800 Universal Access Server.

⁵ Use the interface ethernet router-shelf / slot / port command for a Cisco 7200 VXR used as a router shelf in an AS5800 Universal Access Server.

⁶ Use the interface fastethernet router-shelf / slot / port command for a Cisco 7200 VXR used as a router shelf in an AS5800 Universal Access Server.

⁷ Use the interface gigabitethernet router-shelf / slot / port command for a Cisco 7200 VXR used as a router shelf in an AS5800 Universal Access Server.

Command or Action	Purpose
Step 2 Router(config)# interface ethernet slot / port ⁵ Example:	Selects the Ethernet interface on the I/O controller in slot 0 in port adapter slot 1 to configure.
Step 3 Router(config)# interface fastethernet slot / port ⁶ Example:	Selects the Fast Ethernet interface on the I/O controller in slot 0 in port adapter slot 2 to configure.
Step 4 Router(config)# interface gigabitethernet slot / port ⁷	Selects the Gigabit Ethernet interface on the I/O controller in slot 0 in port adapter slot 0 to configure.
Step 5 Router(config-if) # ip address ip-address mask	Assigns an IP address and subnet mask to the interface (if IP routing is enabled on the system).
Step 6 Router(config-if)# duplex auto	Changes the Fast Ethernet interface port transmission mode to autonegotiation.
Step 7 Router#(config-if)# Exit	Exits configuration mode.

Verifying the Configuration

Use the **show interfaces {ethernet | fastethernet | gigabitethernet}** command to verify that the interface and line protocol are in the correct state (up) and that the transmission mode is configured on the interface. You can configure full, half, or auto transmission mode for Ethernet and Fast Ethernet interfaces. You can configure forced transmission mode for Gigabit Ethernet interfaces. The following is sample output from the **show interfaces gigabitethernet** command.

```
Router# show interfaces gigabitethernet 0/0
GigabitEthernet0/0 is up, line protocol is up
  Hardware is 82543 (Livengood), address is 00d0.ffb6.4c00 (bia 00d0.ffb6.4c00)
  Internet address is 10.1.1.0/0
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex mode, link type is autonegotiation, media type is SX
  output flow-control is on, input flow-control is on
  ARP type:ARPA, ARP Timeout 04:00:00
  Last input 00:00:04, output 00:00:03, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy:fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    2252 packets input, 135120 bytes, 0 no buffer
```

⁵ Use the interface ethernet router-shelf / slot / port command for a Cisco 7200 VXR used as a router shelf in an AS5800 Universal Access Server.

⁶ Use the interface fastethernet router-shelf / slot / port command for a Cisco 7200 VXR used as a router shelf in an AS5800 Universal Access Server.

⁷ Use the interface gigabitethernet router-shelf / slot / port command for a Cisco 7200 VXR used as a router shelf in an AS5800 Universal Access Server.

```

Received 2252 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
0 watchdog, 0 multicast, 0 pause input
0 input packets with dribble condition detected
2631 packets output, 268395 bytes, 0 underruns
0 output errors, 0 collisions, 2 interface resets
0 babbles, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier, 0 pause output
0 output buffer failures, 0 output buffers swapped out

```

Monitoring and Maintaining the Cisco 7200-I O GE+E and Cisco 7200-I O-2FE E

To monitor and maintain the Gigabit Ethernet or Ethernet interfaces on the Cisco 7200-I/O-GE+E, use the following commands in privileged EXEC mode.

Command or Action	Purpose
Router# <code>show controllers ethernet</code>	Displays hardware and software information about the Ethernet interface.
Router# <code>show interfaces ethernet slot/port</code> ⁸	Displays information about the Ethernet interface on the router.
Router# <code>show controllers gigabitethernet</code>	Displays hardware and software information about the Gigabit Ethernet interface.
Router# <code>show interfaces gigabitethernet slot/port</code> ⁹	Displays information about a Gigabit Ethernet interface on the router.

To monitor and maintain the Fast Ethernet or Ethernet interfaces on the Cisco 7200-I/O-2FE/E, use the following commands in privileged EXEC mode.

Command or Action	Purpose
Router# <code>show controllers ethernet</code>	Displays hardware and software information about the Ethernet interface.
Router# <code>show interfaces ethernet slot/port</code> ¹⁰	Displays information about an Ethernet interface on the router.
Router# <code>show controllers fastethernet</code>	Displays hardware and software information about the Fast Ethernet interfaces.
Router# <code>show interfaces fastethernet</code>	Displays information about a Fast Ethernet interface on the router.

⁸ Use the `show interfaces ethernet router-shelf/slot/port` command for a Cisco 7200 VXR used as a router shelf in an AS5800 Universal Access Server

⁹ Use the `interface gigabitethernet router-shelf/slot/port` command for a Cisco 7200 VXR used as a router shelf in an AS5800 Universal Access Server.

¹⁰ Use the `show interfaces ethernet router-shelf/slot/port` command for a Cisco 7200 VXR used as a router shelf in an AS5800 Universal Access Server.

Configuring Fast EtherChannel

The Fast EtherChannel feature allows multiple Fast Ethernet point-to-point links to be bundled into one logical link to provide bidirectional bandwidth of up to 800 Mbps. Fast EtherChannel builds on standards-based 802.3 full-duplex Fast Ethernet to provide fault-tolerant, high-speed links between switches, routers, and servers. This feature can be configured between Cisco 7200 series routers, Cisco 7500 series routers, and Cisco 7000 series routers with the 7000 Series Route Switch Processor (RSP7000) and 7000 Series Chassis Interface (RSP7000CI) or between a Cisco 7500 series router or a Cisco 7000 series router with the RSP7000 and RSP7000CI and a Catalyst 5000 switch.

**Note**

Using the Fast EtherChannel feature on a Catalyst 5000 switch requires a hardware upgrade. Contact your local sales representative for upgrade details.

Fast EtherChannel provides higher bidirectional bandwidth, redundancy, and load sharing. Up to four Fast Ethernet interfaces can be bundled in a port channel, and the router or switch can support up to four port channels. The Fast EtherChannel feature is capable of load balancing traffic across the Fast Ethernet links. Unicast, broadcast, and multicast traffic is distributed across the links providing higher performance and redundant parallel paths. In the event of a link failure, traffic is redirected to remaining links within the Fast EtherChannel without user intervention.

Fast EtherChannel feature, IP traffic is distributed over the port channel interface while traffic from other routing protocols is sent over a single link. Bridged traffic is distributed on the basis of the Layer 3 information in the packet. If the Layer 3 information does not exist in the packet, the traffic is sent over the first link.

Fast EtherChannel supports all features currently supported on the Fast Ethernet interface. You must configure these features on the port-channel interface rather than on the individual Fast Ethernet interfaces. Fast EtherChannel connections are fully compatible with Cisco IOS VLAN and routing technologies. The Inter-Switch Link (ISL) VLAN trunking protocol can carry multiple VLANs across a Fast EtherChannel, and routers attached to Fast EtherChannel links can provide full multiprotocol routing with support for hot standby using Hot Standby Router Protocol (HSRP).

The port channel (consisting of up to four Fast Ethernet interfaces) is treated as a single interface. A port channel is used in the Cisco IOS software to maintain compatibility with existing commands on the Catalyst 5000 switch. You create the Fast EtherChannel by using the **interface port-channel** interface configuration command. You can assign up to four Fast Ethernet interfaces to a port channel by using the **channel-group** interface configuration command.

Additional Fast EtherChannel features include

- Hot Standby Router Protocol (HSRP)

For more information about configuring HSRP, refer to the "Configuring IP Services" chapter in the *Cisco IOS IP Application Services Configuration Guide*.

- Cisco Express Forwarding (CEF) and distributed CEF (dCEF)

For more information about configuring CEF, refer to the "Configuring Cisco Express Forwarding" part of the *Cisco IOS IP Switching Configuration Guide*.

For information on how to configure Ethernet or Fast Ethernet, see the tasks listed in the [Configuring the Ethernet Fast Ethernet and Gigabit Ethernet Interfaces](#), page 29.

- [Fast EtherChannel Configuration Task List](#), page 33

Fast EtherChannel Configuration Task List

To configure Fast EtherChannel, perform the tasks in the following sections. Each task is identified as required or optional.

For information on other commands that can be used by the Fast EtherChannel, refer to the other configuration guides and command references in the Cisco IOS documentation set.

- [Configuring the Port-Channel Interface, page 33](#)
- [Configuring the Fast Ethernet Interfaces, page 34](#)
- [Configuring the Fast Ethernet Interfaces, page 35](#)
- [Configuring the Gigabit Ethernet Interfaces, page 37](#)
- [Configuring the Gigabit Ethernet Interfaces, page 37](#)

Configuring the Port-Channel Interface

To configure the port-channel interface, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. Router(config)# **interface port-channel** *channel-number*
2. Router(config-if)# **ip address** *ip-address mask*
3. Router(config-if)# **mac-address** *ieee-address*
4. Router(config-if)# **end**
5. Router# **show interface port-channel**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config)# interface port-channel <i>channel-number</i>	Creates the port-channel interface and enters interface configuration mode. The channel number can be 1 to 4.
Step 2	Router(config-if)# ip address <i>ip-address mask</i>	Assigns an IP address and subnet mask to the Fast EtherChannel. If you configure ISL, you must assign the IP address to the subinterface (for example, interface port channel 1.1--an IP address per VLAN) and you must specify the encapsulation with VLAN number under that subinterface (for example, encapsulation isl 100).
Step 3	Router(config-if)# mac-address <i>ieee-address</i>	(Optional) Assigns a static MAC address to the Fast EtherChannel. If you do not assign a static MAC address on the port-channel interface, the Cisco IOS software automatically assigns a MAC address. If you assign a static MAC address and then later remove it, the Cisco IOS software automatically assigns a MAC address.
Step 4	Router(config-if)# end	(Optional) Enables other supported interface commands to execute, and exits when they have finished.
Step 5	Router# show interface port-channel	Displays information about the port-channel interface so that you can verify the configuration.

**Note**

If you want to use the Cisco Discovery Protocol (CDP), you must configure it on the physical Ethernet, Fast Ethernet, or Gigabit Ethernet interface, not on the port-channel interface.

**Caution**

With Release 11.1(20)CC and later, Fast EtherChannel supports CEF/dCEF. We recommend that you clear all explicit **ip route-cache distributed** commands from the Fast Ethernet interfaces before enabling dCEF on the port-channel interface. Doing this gives the port-channel interface proper control of its physical Fast Ethernet links. When you enable CEF/dCEF globally, all interfaces that support CEF/dCEF are enabled. When CEF/dCEF is enabled on the port-channel interface, it is automatically enabled on each of the Fast Ethernet interfaces in the channel group. However, if you have previously disabled CEF/dCEF on the Fast Ethernet interface, CEF/dCEF is not automatically enabled. In this case, you must enable CEF/dCEF on the Fast Ethernet interface.

Configuring the Fast Ethernet Interfaces

To assign the Fast Ethernet interfaces to the Fast EtherChannel, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. Router(config)# **interface fastethernet** *slot / port*
2. Router(config-if)# **no ip address**
3. Router(config-if)# **channel-group** *channel-number*
4. Router(config-if)# **exit**
5. Router(config-if)# **end**
6. Router(config)# **show interfaces port-channel**

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 Router(config)# interface fastethernet slot / port</p> <p>Example:</p> <p>(Cisco 7200 series routers)</p> <p>Example:</p> <p>Router(config)# interface fastethernet slot / port-adapter / port</p> <p>Example:</p> <p>(Cisco 7500 series and Cisco 7000 series routers with RSP7000)</p>	Creates or modifies an existing Fast Ethernet interface and enters interface configuration mode.
<p>Step 2 Router(config-if)# no ip address</p>	Disables the IP address before performing the next step, if the Fast Ethernet interface already exists and has an IP address assigned.
<p>Step 3 Router(config-if)# channel-group channel-number</p>	Assigns the Fast Ethernet interfaces to the Fast EtherChannel. The channel number is the same as the channel number that you specified when you created the port-channel interface.
<p>Step 4 Router(config-if)# exit</p>	Exits interface configuration mode. Repeat Steps 1 through 4 to add up to four Fast Ethernet interfaces to the Fast EtherChannel.
<p>Step 5 Router(config-if)# end</p>	(Optional) Enables other supported interface commands to execute, and exits when they have finished.
<p>Step 6 Router(config)# show interfaces port-channel</p>	Displays information about the Fast Ethernet interface so that you can verify the configuration.

Configuring the Fast Ethernet Interfaces

**Caution**

The port-channel interface is the routed interface. Do not enable Layer 3 addresses on the physical Fast Ethernet interfaces. Do not assign bridge groups on the physical Fast Ethernet interfaces because it creates loops. Also, you must disable spanning tree.

To remove a Fast Ethernet interface from a Fast EtherChannel, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. Router(config)# **interface fastethernet slot / port**
2. Router(config-if)# **no channel-group**
3. Router(config-if)# **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config)# interface fastethernet slot / port Example: (Cisco 7200 series routers) Example: Router(config)# interface fastethernet slot / port-adapter / port Example: (Cisco 7500 series and Cisco 7000 series routers with RSP7000)	Specifies the Fast Ethernet interface and enters interface configuration mode.
Step 2	Router(config-if)# no channel-group	Removes the Fast Ethernet interface from the channel group.
Step 3	Router(config-if)# end	(Optional) Enables other supported interface commands to execute, and exits when they have finished.

The Cisco IOS software automatically removes a Fast Ethernet interface from the Fast EtherChannel if the interface goes down, and the software automatically adds the Fast Ethernet interface to the Fast EtherChannel when the interface is back up.

Currently, Fast EtherChannel relies on keepalives to detect whether the line protocol is up or down. Keepalives are enabled by default on the Fast Ethernet interfaces. If the line protocol on the interface goes down because it did not receive a keepalive signal, the Fast EtherChannel detects that the line protocol is down and removes the interface from the Fast EtherChannel. However, if the line protocol remains up because keepalives are disabled on the Fast Ethernet interface, the Fast EtherChannel cannot detect this link failure (other than a cable disconnect) and does not remove the interface from the Fast EtherChannel even if the line protocol goes down. This can result in unpredictable behavior. The implementation of the Port Aggregation Protocol in a subsequent release of this feature will remove the dependency on keepalives.

See the [LAN Interface Configuration Examples, page 51](#) for configuration examples.

You can monitor the status of the Fast EtherChannel interface by using the **show interfaces port-channel EXEC** command.

Configuring the Gigabit Ethernet Interfaces

To assign the Gigabit Ethernet interfaces to the Gigabit EtherChannel, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. Router(config)# **interface gigabitethernet** *slot / port*
2. Router(config-if)# **no ip address**
3. Router(config-if)# **channel-group** *channel-number*
4. Router(config-if)# **exit**
5. Router(config-if)# **end**
6. Router(config)# **show interfaces port-channel**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config)# interface gigabitethernet <i>slot / port</i> Example: (Cisco 7200 series routers)	Creates or modifies an existing Gigabit Ethernet interface and enters interface configuration mode.
Step 2	Router(config-if)# no ip address	Disables the IP address before performing the next step, if the Gigabit Ethernet interface already exists and has an IP address assigned.
Step 3	Router(config-if)# channel-group <i>channel-number</i>	Assigns the Gigabit Ethernet interfaces to the Gigabit EtherChannel. The channel number is the same as the channel number that you specified when you created the port-channel interface.
Step 4	Router(config-if)# exit	Exits interface configuration mode. Repeat Steps 1 through 4 to add up to eight Gigabit Ethernet interfaces to the Gigabit EtherChannel.
Step 5	Router(config-if)# end	(Optional) Enables other supported interface commands to execute, and exits when they have finished.
Step 6	Router(config)# show interfaces port-channel	Displays information about the Gigabit Ethernet interface so that you can verify the configuration.

Configuring the Gigabit Ethernet Interfaces

To remove a Gigabit Ethernet interface from a Gigabit EtherChannel, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. Router(config)# **interface gigabitethernet** *slot / port*
2. Router(config-if)# **no channel-group**
3. Router(config-if)# **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config)# interface gigabitethernet slot / port Example: (Cisco 7200 series routers)	Specifies the Gigabit Ethernet interface and enters interface configuration mode.
Step 2	Router(config-if)# no channel-group	Removes the Gigabit Ethernet interface from the channel group.
Step 3	Router(config-if)# end	(Optional) Enables other supported interface commands to execute, and exits when they have finished.

You can monitor the status of the Gigabit EtherChannel interface by using the **show interfaces port-channel EXEC** command.

Configuring a FDDI Interface

The FDDI is an ANSI-defined standard for timed 100-Mbps token passing over fiber-optic cable. FDDI is not supported on access servers.

A FDDI network consists of two counter-rotating, token-passing fiber-optic rings. On most networks, the primary ring is used for data communication and the secondary ring is used as a hot standby. The FDDI standard sets a total fiber length of 200 kilometers. (The maximum circumference of the FDDI network is only half the specified kilometers because of the wrapping or looping back of the signal that occurs during fault isolation.)

The FDDI standard allows a maximum of 500 stations with a maximum distance between active stations of 2 kilometers when interconnecting them with multimode fiber or 10 kilometers when interconnected via single mode fiber, both of which are supported by our FDDI interface controllers. The FDDI frame can contain a minimum of 17 bytes and a maximum of 4500 bytes. Our implementation of FDDI supports Station Management (SMT) Version 7.3 of the X3T9.5 FDDI specification, offering a single MAC dual-attach interface that supports the fault-recovery methods of the dual attachment stations (DASs). The mid-range platforms also support single attachment stations (SASs).

Refer to the *Cisco Product Catalog* for specific information on platform and interface compatibility. For installation and configuration information, refer to the installation and configuration publication for the appropriate interface card or port adapter.

- [Source-Route Bridging over FDDI on Cisco 4000-M Cisco 4500-M and Cisco 4700-M Routers, page 39](#)
- [Particle-Based Switching of Source-Route Bridge Packets on Cisco 7200 Series Routers, page 39](#)
- [Using Connection Management Information, page 39](#)
- [FDDI Configuration Task List, page 40](#)

Source-Route Bridging over FDDI on Cisco 4000-M Cisco 4500-M and Cisco 4700-M Routers

Source-route bridging (SRB) is supported on the FDDI interface to the Cisco 4000-M, Cisco 4500-M, and Cisco 4700-M routers. For instructions on configuring autonomous FDDI SRB or fast-switching SRB over FDDI, refer to the "Configuring Source-Route Bridging" chapter of the *CiscoIOS Bridging and IBM Networking Configuration Guide*.

Particle-Based Switching of Source-Route Bridge Packets on Cisco 7200 Series Routers

SRB is supported over FDDI. Particle-based switching is supported for SRB packets (over FDDI and Token Ring) by default.

Particle-based switching adds scatter-gather capability to SRB to improve performance. Particles represent a communications data packet as a collection of noncontiguous buffers. The traditional Cisco IOS packet has a packet type control structure and a single contiguous data buffer. A particle packet has the same packet type control structure, but also maintains a queue of particle type structures, each of which manages its own block.

The scatter-gather architecture used by particle-based switching provides the following advantages:

- Allows drivers to use memory more efficiently (especially when using media that has a large maximum transmission unit [MTU]). For example, Token Ring buffers could be 512 bytes rather than 16 KB.
- Allows concurrent use of the same region of memory. For example, on IP multicast a single packet is received and sent out on multiple interfaces simultaneously.
- Allows insertion or deletion of memory at any location in a packet (not just at the beginning or end).

For information about configuring SRB over FDDI, refer to the "Configuring Source-Route Bridging" chapter of the *CiscoIOS Bridging and IBM Networking Configuration Guide*.

Using Connection Management Information

Connection management (CMT) is a FDDI process that handles the transition of the ring through its various states (off, on, active, connect, and so on) as defined by the X3T9.5 specification. The FIP (FDDI Interface Processor) provides CMT functions in microcode.

A partial sample output of the **show interfaces fddi** command follows, along with an explanation of how to interpret the CMT information in the output.

```
Phy-A state is active, neighbor is B, cmt signal bits 08/20C, status ALS
Phy-B state is active, neighbor is A, cmt signal bits 20C/08, status ILS
CFM is thru A, token rotation 5000 usec, ring operational 0:01:42
Upstream neighbor 0800.2008.C52E, downstream neighbor 0800.2008.C52E
```

The **show interfaces fddi** example shows that Physical A (Phy-A) completed CMT with its neighbor. The state is active, and the display indicates a Physical B-type neighbor.

The sample output indicates CMT signal bits 08/20C for Phy-A. The transmit signal bits are 08. Looking at the pulse code modulation (PCM) state machine, 08 indicates that the port type is A, that the port compatibility is set, and that the LCT duration requested is short. The receive signal bits are 20C, that indicate that the neighbor type is B, that port compatibility is set, that there is a MAC on the port output, and so on.

The neighbor is determined from the received signal bits, as follows:

Bit Positions	9 8 7 6 5 4 3 2 1 0
Value Received	1 0 0 0 0 0 1 1 0 0

Interpreting the bits in the diagram above, the received value equals 0x20C. Bit positions 1 and 2 (0 1) indicate a Physical B-type connection.

The transition states displayed indicate that the CMT process is running and actively trying to establish a connection to the remote physical connection. The CMT process requires state transition with different signals being transmitted and received before moving on to the state ahead as indicated in the PCM state machine. The 10 bits of CMT information are transmitted and received in the Signal State. The NEXT state is used to separate the signaling performed in the Signal State. Therefore, in the preceding sample output, the NEXT state was entered 11 times.



Note

The display line showing transition states is not generated if the FDDI interface has been shut down, or if the **cmt disconnect** command has been issued, or if the **fdi if-cmt** command has been issued. (The **fdi if-cmt** command applies to the Cisco 7500 series routers only.)

The CFM state is through A in the sample output, which means the Phy-A of this interface has successfully completed CMT with the Phy-B of the neighbor and Phy-B of this interface has successfully completed CMT with the Phy-A of the neighbor.

The display (or nondisplay) of the upstream and downstream neighbor does not affect the ability to route data. Because the upstream neighbor is also its downstream neighbor in the sample, there are only two stations in the ring: the network server and the router at address 0800.2008.C52E.

FDDI Configuration Task List

To configure a FDDI interface, perform the tasks in the following sections. Each task in the list is identified as either required or optional.

- [Specifying a FDDI Interface, page 41](#)
- [Enabling FDDI Bridging Encapsulation, page 41](#)
- [Enabling Full-Duplex Mode on the FDDI Interface, page 41](#)
- [Setting the Token Rotation Time, page 42](#)
- [Setting the Transmission Valid Timer, page 42](#)
- [Controlling the Transmission Timer, page 42](#)
- [Modifying the C-Min Timer, page 43](#)
- [Modifying the TB-Min Timer, page 43](#)
- [Modifying the FDDI Timeout Timer, page 43](#)
- [Controlling SMT Frame Processing, page 43](#)
- [Enabling Duplicate Address Checking, page 43](#)
- [Setting the Bit Control, page 44](#)
- [Controlling the CMT Microcode, page 44](#)
- [Starting and Stopping FDDI, page 44](#)
- [Setting FDDI Frames per Token Limit, page 45](#)
- [Controlling the FDDI SMT Message Queue Size, page 45](#)
- [Preallocating Buffers for Bursty FDDI Traffic, page 45](#)

Specifying a FDDI Interface

To specify a FDDI interface and enter interface configuration mode, use one of the following commands in global configuration mode.

Command or Action	Purpose
<code>Router(config)# interface fddi number</code>	Enters interface configuration.
<code>Router(config)# interface fddi slot / port</code>	Enters interface configuration for the Cisco 7200 or Cisco 7500 series routers.

Enabling FDDI Bridging Encapsulation

By default, Cisco FDDI uses the SNAP encapsulation format defined in RFC 1042. It is not necessary to define an encapsulation method for this interface when using the FIP.

FIP fully supports transparent and translational bridging for the following configurations:

- FDDI-to-FDDI
- FDDI-to-Ethernet
- FDDI-to-Token Ring

Enabling FDDI bridging encapsulation places the FIP into encapsulation mode when doing bridging. In transparent mode, the FIP interoperates with earlier versions of encapsulating interfaces when performing bridging functions on the same ring. When using the FIP, you can specify the encapsulation method by using the following command in interface configuration mode.

Command or Action	Purpose
<code>Router(config-if)# fddi encapsulate</code>	Specifies the encapsulation method for the FIP.

When you are doing translational bridging, use routing for routable protocols and use translational bridging for the rest, such as local-area transport (LAT).



Note

Bridging between dissimilar media presents several problems that can prevent communications. These problems include bit-order translation (using MAC addresses as data), maximum transfer unit (MTU) differences, frame status differences, and multicast address usage. Some or all of these problems might be present in a multimedia-bridged LAN and might prevent communication. These problems are most prevalent in networks that bridge between Token Ring and Ethernet networks or between Token Ring and FDDI because of the different ways that Token Ring is implemented by the end nodes.

We are currently aware of problems with the following protocols when bridged between Token Ring and other media: AppleTalk, DECnet, IP, Novell IPX, Phase IV, VINES, and XNS. Further, the following protocols might have problems when bridged between FDDI and other media: Novell IPX and XNS. We recommend that these protocols be routed whenever possible.

Enabling Full-Duplex Mode on the FDDI Interface

To enable full-duplex mode on the PA-F/FD-SM and PA-F/FD-MM port adapters, use one of the following commands in interface configuration mode.

Command or Action	Purpose
Router(config-if)# full-duplex	Enables full-duplex on the FDDI interface of the PA-F/FD-SM and PA-F/FD-MM port adapter.
or	
Router(config-if)# no half-duplex	

Setting the Token Rotation Time

You can set the FDDI token rotation time to control ring scheduling during normal operation and to detect and recover from serious ring error situations. To do so, use the following command in interface configuration mode.

Command or Action	Purpose
Router(config-if)# fdi token-rotation-time <i>microseconds</i>	Sets the FDDI token rotation time.

The FDDI standard restricts the allowed time to greater than 4000 microseconds and less than 165,000 microseconds. As defined in the X3T9.5 specification, the value remaining in the token rotation timer (TRT) is loaded into the token holding timer (THT). Combining the values of these two timers provides the means to determine the amount of bandwidth available for subsequent transmissions.

Setting the Transmission Valid Timer

You can set the transmission timer to recover from a transient ring error by using the following command in interface configuration mode.

Command or Action	Purpose
Router(config-if)# fdi valid-transmission-time <i>microseconds</i>	Sets the FDDI valid transmission timer.

Controlling the Transmission Timer

You can set the FDDI control transmission timer to control the FDDI TL-Min time, which is the minimum time to transmit a Physical Sublayer or PHY line state before advancing to the next Physical Connection Management or PCM state as defined by the X3T9.5 specification. To do so, use the following command in interface configuration mode.

Command or Action	Purpose
Router(config-if)# fdi tl-min-time <i>microseconds</i>	Sets the FDDI control transmission timer.

Modifying the C-Min Timer

You can modify the C-Min timer on the PCM from its default value of 1600 microseconds by using the following command in interface configuration mode.

Command or Action	Purpose
<code>Router(config-if)# fdi c-min <i>microseconds</i></code>	Sets the C-Min timer on the PCM.

Modifying the TB-Min Timer

You can change the TB-Min timer in the PCM from its default value of 100 milliseconds. To do so, use the following command in interface configuration mode.

Command or Action	Purpose
<code>Router(config-if)# fdi tb-min <i>milliseconds</i></code>	Sets TB-Min timer in the PCM.

Modifying the FDDI Timeout Timer

You can change the FDDI timeout timer in the PCM from its default value of 100 ms. To do so, use the following command in interface configuration mode.

Command or Action	Purpose
<code>Router(config-if)# fdi t-out <i>milliseconds</i></code>	Sets the timeout timer in the PCM.

Controlling SMT Frame Processing

You can disable and enable SMT frame processing for diagnostic purposes. To do so, use one of the following commands in interface configuration mode.

Command or Action	Purpose
<code>Router(config-if)# no fdi smt-frames</code>	Disables SMT frame processing.
<code>Router(config-if)# fdi smt-frames</code>	Enables SMT frame processing.

Enabling Duplicate Address Checking

You can enable the duplicate address detection capability on the FDDI. If the FDDI finds a duplicate address, it displays an error message and shuts down the interface. To enable duplicate address checking, use the following command in interface configuration mode.

Command or Action	Purpose
Router(config-if)# fdi duplicate-address-check	Enables duplicate address checking capability.

Setting the Bit Control

You can set the FDDI bit control to control the information transmitted during the Connection Management (CMT) signaling phase. To do so, use the following command in interface configuration mode.

Command or Action	Purpose
Router(config-if)# fdi cmt-signal-bits <i>signal-bits</i> [phy-a phy-b]	Sets the FDDI bit control.

Controlling the CMT Microcode

You can control whether the CMT onboard functions are on or off. The FIP provides CMT functions in microcode. These functions are separate from those provided on the processor card and are accessed through EXEC commands.

The default is for the FIP CMT functions to be on. A typical reason to disable these functions is when you work with new FDDI equipment and have problems bringing up the ring. If you disable the CMT microcode, the following actions occur:

- The FIP CMT microcode is disabled.
- The main system code performs the CMT function while debugging output is generated.

To disable the CMT microcode, use the following command in interface configuration mode.

Command or Action	Purpose
Router(config-if)# no fdi if-cmt	Disables the FCIT CMT functions.

Starting and Stopping FDDI

In normal operation, the FDDI interface is operational once the interface is connected and configured. You can start and stop the processes that perform the CMT function and allow the ring on one fiber to be stopped. To do so, use either of the following commands in EXEC mode.

Command or Action	Purpose
Router# cmt connect [<i>interface-name</i> [phy-a phy-b]]	Starts CMT processes on a FDDI ring.
Router# cmt disconnect [<i>interface-name</i> [phy-a phy-b]]	Stops CMT processes on a FDDI ring.

Do not use either of the preceding commands during normal operation of FDDI; they are used during interoperability tests.

Setting FDDI Frames per Token Limit

The FDDI interface is able to transmit multiple frames per token on a Cisco 4000, a Cisco 4500, and a Cisco 4700 series router, instead of transmitting only a single frame at a time. You can specify the maximum number of frames to be transmitted with each token capture. This significantly improves your throughput when you have heavy or very bursty traffic.

To configure the FDDI interface to transmit a maximum number of frames per token capture, use the following commands beginning in privileged EXEC mode.

SUMMARY STEPS

1. Router# **configure terminal**
2. Router(config)# **interface fddi number**
3. Router(config-if)# **fddi number**
4. Router(config-if)# **fddi frames-per-token number**
5. Router(config-if)# **fddi frames-per-token number**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# interface fddi number	Enters interface configuration mode.
Step 3	Router(config-if)# fddi number	Shows fddi command options.
Step 4	Router(config-if)# fddi frames-per-token number	Shows fddi frames-per-token command options.
Step 5	Router(config-if)# fddi frames-per-token number	Specifies the maximum number of frames to be transmitted per token capture.

Controlling the FDDI SMT Message Queue Size

You can set the maximum number of unprocessed FDDI Station Management (SMT) frames that will be held for processing. Setting this number is useful if the router that you are configuring gets bursts of messages that arrive faster than the router can process. To set the number of frames, use the following command in global configuration mode.

Command or Action	Purpose
Router(config)# smt-queue-threshold number	Sets SMT message queue size.

Preallocating Buffers for Bursty FDDI Traffic

The FCI card preallocates three buffers to handle bursty FDDI traffic (for example, Network File System (NFS) bursty traffic). You can change the number of preallocated buffers use the following command in interface configuration mode.

Command or Action	Purpose
Router(config-if)# fdi burst-count	Preallocates buffers to handle bursty FDDI traffic.

Configuring a Hub Interface

Cisco 2500 series includes routers that have hub functionality for an Ethernet interface. The hub is a multiport repeater. The advantage of an Ethernet interface over a hub is that the hub provides a star-wiring physical network configuration while the Ethernet interface provides 10BASE-T physical network configuration. The router models with hub ports and their configurations are as follows:

- Cisco 2505--1 Ethernet (8 ports) and 2 serial
- Cisco 2507--1 Ethernet (16 ports) and 2 serial
- Cisco 2516--1 Ethernet (14 ports), 2 serial, and 1 ISDN BRI

Cisco provides Simple Network Management Protocol (SNMP) management of the Ethernet hub as specified in RFC 1516, *Definitions of Managed Objects for IEEE 802.3 Repeater Devices*.

To configure hub functionality on an Ethernet interface, perform the tasks in the following sections. Each task in the list is identified as either required or optional.

For configuration examples, see the [Hub Configuration Examples](#), page 56.

- [Enabling a Hub Port](#), page 46
- [Disabling or Enabling Automatic Receiver Polarity Reversal](#), page 46
- [Disabling or Enabling the Link Test Function](#), page 47
- [Enabling Source Address Control](#), page 47
- [Enabling SNMP Illegal Address Trap](#), page 48

Enabling a Hub Port

To enable a hub port, use the following commands in global configuration mode.

SUMMARY STEPS

1. Router(config)# **hub ethernet** *numberport* [*end-port*]
2. Router(config)# **no shutdown**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config)# hub ethernet <i>numberport</i> [<i>end-port</i>]	Specifies the hub number and the hub port (or range of hub ports) and enters hub configuration mode.
Step 2	Router(config)# no shutdown	Enables the hub ports.

Disabling or Enabling Automatic Receiver Polarity Reversal

On Ethernet hub ports only, the hub ports can invert, or correct, the polarity of the received data if the port detects that the received data packet waveform polarity is reversed because of a wiring error. This receive

circuitry polarity correction allows the hub to repeat subsequent packets with correct polarity. When enabled, this function is executed once after reset of a link fail state.

Automatic receiver polarity reversal is enabled by default. To disable this feature on a per-port basis, use the following command in hub configuration mode.

Command or Action	Purpose
<code>Router(config-hub)# no auto-polarity</code>	Disables automatic receiver polarity reversal.

To enable automatic receiver polarity reversal on a per-port basis, use the following command in hub configuration mode.

Command or Action	Purpose
<code>Router(config-hub)# auto-polarity</code>	Enables automatic receiver polarity reversal.

Disabling or Enabling the Link Test Function

The link test function applies to Ethernet hub ports only. The Ethernet ports implement the link test function as specified in the 802.3 10BASE-T standard. The hub ports will transmit link test pulses to any attached twisted pair device if the port has been inactive for more than 8 to 17 milliseconds.

If a hub port does not receive any data packets or link test pulses for more than 65 to 132 milliseconds and the link test function is enabled for that port, that port enters link fail state and cannot transmit or receive. The hub port is enabled again when it receives four consecutive link test pulses or a data packet.

The link test function is enabled by default. To allow the hub to interoperate with 10BASE-T twisted-pair networks that do not implement the link test function, the link test receive function of the hub can be disabled on a per-port basis. To do so, use the following command in hub configuration mode.

Command	Purpose
<code>Router(config-hub)# no link-test</code>	Disables the link test function.

To enable the link test function on a hub port connected to an Ethernet interface, use the following command in hub configuration mode.

Command	Purpose
<code>Router(config-hub)# link-test</code>	Enables the link test function.

Enabling Source Address Control

On an Ethernet hub port only, you can configure a security measure such that the port accepts packets only from a specific MAC address. For example, suppose your workstation is connected to port 3 on a hub, and source address control is enabled on port 3. Your workstation has access to the network because the hub accepts any packet from port 3 with the MAC address of the workstation. Any packets that arrive with a different MAC address cause the port to be disabled. The port is enabled again after 1 minute, and the MAC address of incoming packets is checked again.

To enable source address control on a per-port basis, use the following command in hub configuration mode.

Command	Purpose
Router(config-hub)# source-address [<i>mac-address</i>]	Enables source address control.

If you omit the optional MAC address, the hub remembers the first MAC address that it receives on the selected port and allows only packets from the learned MAC address.

See the examples of establishing source address control in the [Hub Configuration Examples, page 56](#).

Enabling SNMP Illegal Address Trap

To enable the router to issue an SNMP trap when an illegal MAC address is detected on an Ethernet hub port, use the following commands in hub configuration mode.

SUMMARY STEPS

1. Router(config-hub)# **hub ethernet** *number port* [*end-port*]
2. Router(config-hub)# **snmp trap illegal-address**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config-hub)# hub ethernet <i>number port</i> [<i>end-port</i>]	Specifies the hub number and the hub port (or range of hub ports) and enters hub configuration mode.
Step 2	Router(config-hub)# snmp trap illegal-address	Enables the router to issue an SNMP trap when an illegal MAC address is detected on the hub port.

You may need to set up a host receiver for this trap type (snmp-server host) for a Network Management System (NMS) to receive this trap type. The default is no trap. For an example of configuring a SNMP trap for an Ethernet hub port, see the [Hub Configuration Examples, page 56](#).

Configuring a Token Ring Interface

Cisco supports various Token Ring interfaces. Refer to the *Cisco Product Catalog* for information about platform and hardware compatibility.

The Token Ring interface supports both routing (Layer 3 switching) and source-route bridging (Layer 2 switching) on a per-protocol basis. For example, IP traffic could be routed, while SNA traffic is bridged. Routing features enhance source-route bridges.

The Token Ring MIB variables support the specification in RFC 1231, *IEEE 802.5 Token Ring MIB*. The mandatory Interface Table and Statistics Table are implemented, but the optional Timer Table of the Token Ring MIB is not. The Token Ring MIB has been implemented for the Token Ring Interface Processor (TRIP).

Use the **show interfaces**, **show controllers token**, and **show controllers cbusEXEC** commands to display the Token Ring numbers. These commands provide a report for each ring that Cisco IOS software supports.

**Note**

If the system receives an indication of a cabling problem from a Token Ring interface, it puts that interface into a reset state and does not attempt to restart it. It functions this way because periodic attempts to restart the Token Ring interface drastically affect the stability of routing tables. Once you have plugged the cable into the MAU (media attachment unit) again, restart the interface by using the **clear interface tokenring number** command, where the *number* argument is the interface number.

By default, the Token Ring interface uses the SNAP encapsulation format defined in RFC 1042. It is not necessary to define an encapsulation method for this interface.

- [Particle-Based Switching of Source-Route Bridge Packets on Cisco 7200 Series Routers, page 49](#)
- [Dedicated Token Ring Port Adapter, page 49](#)
- [Token Ring Interface Configuration Task List, page 50](#)

Particle-Based Switching of Source-Route Bridge Packets on Cisco 7200 Series Routers

Particle-based switching is supported for SRB packets (over FDDI and Token Ring) by default.

Particle-based switching adds scatter-gather capability to SRB to improve performance. Particles represent a communications data packet as a collection of noncontiguous buffers. The traditional Cisco IOS packet has a packet type control structure and a single contiguous data buffer. A particle packet has the same packet type control structure, but it also maintains a queue of particle type structures, each of which manages its own block.

The scatter-gather architecture used by particle-based switching provides the following advantages:

- Allows drivers to use memory more efficiently (especially when using media that has a large maximum transmission unit [MTU]). For example, Token Ring buffers could be 512 bytes rather than 16 KB.
- Allows concurrent use of the same region of memory. For example, on IP multicast a single packet is received and sent out on multiple interfaces simultaneously.
- Allows insertion or deletion of memory at any location in a packet (not just at the beginning or end).

For information about configuring SRB over FDDI, refer to the "Configuring Source-Route Bridging" chapter of the *CiscoIOS Bridging and IBM Networking Configuration Guide*.

Dedicated Token Ring Port Adapter

The Dedicated Token Ring port adapter (PA-4R-DTR) is available on Cisco 7500 series routers, Cisco 7200 series routers, and Cisco 7000 series routers with the 7000 Series Route Switch Processor (RSP7000) and 7000 Series Chassis Interface (RSP7000CI).

The PA-4R-DTR provides up to four IBM Token Ring or IEEE 802.5 Token Ring interfaces. Each Token Ring interface can be set for 4-Mbps or 16-Mbps half-duplex or full-duplex operation and can operate as a standard Token Ring station or as a concentrator port. The default for all interfaces is Token Ring station mode with half-duplex 16-Mbps operation. The PA-4R-DTR connects over Type 1 lobe or Type 3 lobe cables, with each interface providing an RJ-45 receptacle.

Token Ring Interface Configuration Task List

To configure a Token Ring interface, perform the tasks in the following sections. Each task is identified as either required or optional.

- [Specifying a Token Ring Interface, page 50](#)
- [Enabling Early Token Release, page 50](#)
- [Configuring PCbus Token Ring Interface Management, page 50](#)
- [Enabling a Token Ring Concentrator Port, page 51](#)
- [Monitoring and Maintaining the Port, page 51](#)

Specifying a Token Ring Interface

To specify a Token Ring interface and enter interface configuration mode, use one of the following commands in global configuration mode.

Command	Purpose
Router(config)# interface tokenring <i>number</i>	Enters interface configuration mode.
Router(config)# interface tokenring <i>slot / port</i>	Enters interface configuration mode for the Cisco 7200 or Cisco 7500 series routers.
Router(config)# interface tokenring <i>slot / port-adaptor / port</i>	Enters interface configuration mode for the Cisco 7500 series routers.

Enabling Early Token Release

Cisco Token Ring interfaces support early token release, a method whereby the interface releases the token back onto the ring immediately after transmitting rather than waiting for the frame to return. This feature can help to increase the total bandwidth of the Token Ring. To configure the interface for early token release, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# early-token-release	Enables early token release.

Configuring PCbus Token Ring Interface Management

The Token Ring interface on the AccessPro PC card can be managed by a remote LAN manager over the PCbus interface. Currently, the LanOptics Hub Networking Management software running on an IBM-compatible PC is supported.

To enable LanOptics Hub Networking Management of a PCbus Token Ring interface, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# local-lnm	Enables PCbus LAN management.

Enabling a Token Ring Concentrator Port

To enable an interface to operate as a concentrator port, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# port	Specifies concentrator port operation.

Monitoring and Maintaining the Port

To monitor the Token Ring concentrator port, use one or more of the following commands in EXEC mode.

Command	Purpose
Router# show controllers token	Displays internal state information about the Token Ring interfaces in the system.
Router# show interfaces token	Displays high-level statistics for a particular interface.

LAN Interface Configuration Examples

This section provides the following examples to illustrate configuration tasks described in this chapter.

- [Ethernet Encapsulation Enablement Example, page 51](#)
- [Full-Duplex Enablement Operation Example, page 52](#)
- [PA-12E 2FE Port Configuration Examples, page 52](#)
- [PA-VG100 Port Adapter Configuration Example, page 53](#)
- [Cisco 7200-I O-GE+E and Cisco 7200-I O-2FE E Configuration Examples, page 53](#)
- [Fast EtherChannel Configuration Examples, page 54](#)
- [FDDI Frames Configuration Example, page 56](#)
- [Hub Configuration Examples, page 56](#)

Ethernet Encapsulation Enablement Example

These commands enable standard Ethernet Version 2.0 encapsulation on the Ethernet interface processor in slot 4 on port 2 of a Cisco 7500 series router:

```
interface ethernet 4/2
 encapsulation arpa
```

Full-Duplex Enablement Operation Example

The following example assigns an IP address and subnet mask, specifies an MII Ethernet connector, and enables full-duplex mode on Fast Ethernet interface port 0 in slot 1 port adapter 0:

```
Router(config)# interface fastethernet 1/0/0
Router(config-if)#
ip address 10.1.1.10 255.255.255.0
Router(config-if)#
full-duplex
Router(config-if)#
media-type mii
Router(config-if)# exit
Router(config)# exit
```

PA-12E 2FE Port Configuration Examples

The following is an example of a configuration for the PA-12E/2FE port adapter interface. Bridge groups 10, 20, and 30 use IEEE Spanning Tree Protocol. The first four interfaces of a PA-12E/2EF port adapter in port adapter slot 3 use bridge groups 10 and 20. Each interface is assigned to a bridge group, and the shutdown state is set to up. The PA-12E/2FE port adapter supports store-and-forward or cut-through switching technology between interfaces within the same bridge group; store-and-forward is the default. In the following example, the **cut-through** command is used to configure each interface for cut-through switching of received and transmitted data:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL-Z.
Router(config)#
bridge 10 protocol ieee
Router(config)#
bridge 20 protocol ieee
Router(config)#
bridge 30 protocol ieee
Router(config)#

interface fastethernet 3/0
Router(config-if)#

bridge-group 10
Router(config-if)#

cut-through
Router(config-if)#
no shutdown
Router(config-if)#
exit
Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Fast Ethernet3/0, changed
state to up
%LINK-3-UPDOWN: Interface Fast Ethernet3/0, changed state to up
Router(config)#
interface fastethernet 3/1
Router(config-if)#
bridge-group 10
Router(config-if)#

cut-through
Router(config-if)#

no shutdown
Router(config-if)#
exit
Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Fast Ethernet3/1, changed
state to up
```

```

%LINK-3-UPDOWN: Interface Fast Ethernet3/1, changed state to up
Router(config)#
  interface ethernet 3/2
Router(config-if)#
  bridge-group 20
Router(config-if)#
  cut-through
Router(config-if)#

no shutdown
Router(config-if)#
exit
Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet3/2, changed state to up
%LINK-3-UPDOWN: Interface Ethernet3/2, changed state to up
Router(config)#
  interface ethernet 3/3
Router(config-if)#
  bridge-group 20
Router(config-if)#
  cut-through
Router(config-if)#

no shutdown
Router(config-if)#
exit
Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet3/3, changed state to up
%LINK-3-UPDOWN: Interface Ethernet3/3, changed state to up

```

The following example shows integrated routing and bridging enabled on the bridge groups. Bridge group 10 is assigned an IP address and subnet mask, and the shutdown state is changed to up. Bridge group 10 is configured to route IP.

```

Router(config)#
bridge irb
Router(config)#
interface bvi 10
Router(config-if)#
ip address 10.1.15.1 255.255.255.0
Router(config-if)#
no shutdown
Router(config-if)#
exit
Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface BVI10, changed state to up
Router(config)#
bridge 10 route ip
Router(config)# exit
Router#

```

PA-VG100 Port Adapter Configuration Example

The following is an example of a basic configuration for the PA-VG100 port adapter interface in slot 1 on a Cisco 7500 series router. In this example, IP routing is enabled on the router, so an IP address and subnet mask are assigned to the interface.

```

configure terminal
interface vg-anylan 1/0/0
 ip address 10.1.1.10 255.255.255.0
 no shutdown
 exit
exit

```

Cisco 7200-I O-GE+E and Cisco 7200-I O-2FE E Configuration Examples

This section provides the following configuration examples:

- [Configuring the Gigabit Ethernet Interface on the Cisco 7200-I O-GE+E](#), page 54
- [Configuring Autonegotiation on the Cisco 7200-I O-2FE E](#), page 54

Configuring the Gigabit Ethernet Interface on the Cisco 7200-I O-GE+E

The following example configures the Gigabit Ethernet interface on the Cisco 7200-I/O-GE+E. The following commands are configured on slot 0, port 0.

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface gigabitethernet 0/0
Router(config-if)# ip address 10.1.1.10 255.255.255.252
Router(config-if)# negotiation auto
Router(config-if)# end
```

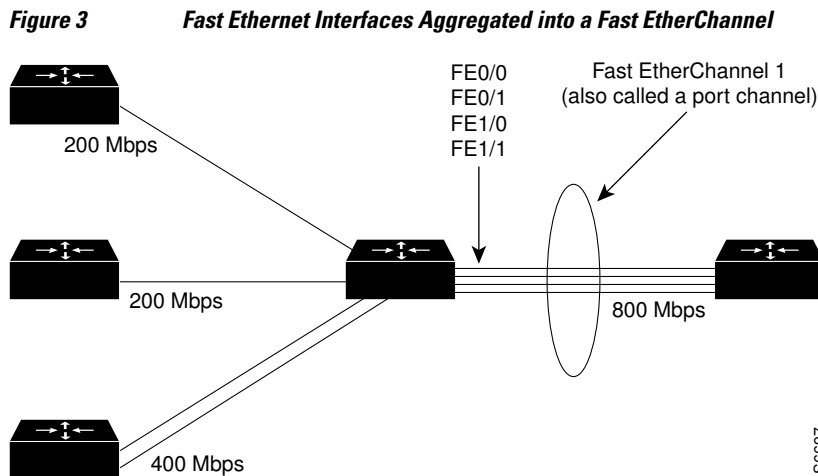
Configuring Autonegotiation on the Cisco 7200-I O-2FE E

The following example configures the Fast Ethernet interface on the Cisco 7200-I/O-2FE/E for fully enabled autonegotiation:

```
Router#
configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface fastethernet 0/0
Router(config-if)#
duplex auto
Router(config-if)# speed auto
```

Fast EtherChannel Configuration Examples

The figure below shows four point-to-point Fast Ethernet interfaces that are aggregated into a single Fast EtherChannel interface.



The configuration file that illustrates this topology follows.

The following is an example of how to create a Fast EtherChannel (port-channel interface) with four Fast Ethernet interfaces. In this example, ISL is enabled on the Fast EtherChannel, and an IP address is assigned to the subinterface.

```
Router# configure terminal
```

```

Router(config)# interface port-channel 1
Router(config-if)# no shutdown
Router(config-if)# exit
Router(config)# interface port-channel 1.1
Router(config-if)# ip address 10.1.1.10 255.255.255.0
Router(config-if)# encapsulation isl 100
Router(config-if)# exit
Router(config)# interface fastethernet 0/0/0
Router(config-if)# no ip address
Router(config-if)# channel-group 1
Fast Ethernet 0/0 added as member-1 to port-channell.
Router(config-if)# exit
Router(config)# interface fastethernet 0/1/0
Router(config-if)# no ip address
Router(config-if)# channel-group 1
Fast Ethernet 0/1 added as member-2 to port-channell.
Router(config-if)# exit
Router(config)# interface fastethernet 1/0/0
Router(config-if)# no ip address
Router(config-if)# channel-group 1
Fast Ethernet 1/0 added as member-3 to port-channell.
Router(config-if)# exit
Router(config)# interface fastethernet 1/1/0
Router(config-if)# no ip address
Router(config-if)# channel-group 1
Fast Ethernet 1/1 added as member-4 to port-channell.
Router(config-if)# exit
Router(config)# exit
Router#

```

The following is a partial example of a configuration file. The MAC address is automatically added to the Fast Ethernet interface when the interfaces are added to the Fast EtherChannel.



Note

If you do not assign a static MAC address on the port-channel interface, the Cisco IOS software automatically assigns a MAC address. If you assign a static MAC address and then later remove it, the Cisco IOS software automatically assigns a MAC address.

```

interface Port-channell
 ip address 10.1.1.10 255.255.255.0
 !
interface Port-channell.1
 encapsulation isl 100
 !
interface Fast Ethernet0/0/0
 mac-address 00e0.1476.7600
 no ip address
 channel-group 1
 !
interface Fast Ethernet0/1/0
 mac-address 00e0.1476.7600
 no ip address
 channel-group 1
 !
interface Fast Ethernet1/0/0
 mac-address 00e0.1476.7600
 no ip address
 channel-group 1
 !
interface Fast Ethernet1/1/0
 mac-address 00e0.1476.7600
 no ip address
 channel-group 1

```

FDDI Frames Configuration Example

The following example shows how to configure the FDDI interface to transmit four frames per token capture:

```
! Enter global configuration mode.
4700#
configure terminal
! Enter interface configuration mode.
4700(config)#
interface fddi 0
! Show the fddi command options.
4700(config-if)#
fddi ?
  encapsulate           Enable FDDI Encapsulation bridging
  frames-per-token      Maximum frames to transmit per service opportunity
  tl-min-time           Line state transmission time
  token-rotation-time   Set the token rotation timer
  valid-transmission-time Set transmission valid timer
! Show fddi frames-per-token command options.
4700(config-if)#
fddi frames-per-token ?
<1-10> Number of frames per token, default = 3
! Specify 4 as the maximum number of frames to be transmitted per token.
4700(config-if)#
fddi frames-per-token 4
```

Hub Configuration Examples

This section provides the following hub configuration examples:

- [Hub Port Startup Examples, page 56](#)
- [Source Address for an Ethernet Hub Port Configuration Examples, page 56](#)
- [Hub Port Shutdown Examples, page 57](#)
- [SNMP Illegal Address Trap Enablement for Hub Port Example, page 57](#)

Hub Port Startup Examples

The following example configures port 1 on hub 0 of Ethernet interface 0:

```
hub ethernet 0 1
no shutdown
```

The following example configures ports 1 through 8 on hub 0 of Ethernet interface 0:

```
hub ethernet 0 1 8
no shutdown
```

Source Address for an Ethernet Hub Port Configuration Examples

The following example configures the hub to allow only packets from MAC address 1111.2222.3333 on port 2 of hub 0:

```
hub ethernet 0 2
source-address 1111.2222.3333
```

The following example configures the hub to remember the first MAC address received on port 2 and allow only packets from that learned MAC address:

```
hub ethernet 0 2
source-address
```

Hub Port Shutdown Examples

The following example shuts down ports 3 through 5 on hub 0:

```
hub ethernet 0 3 5
shutdown
```

The following example shuts down port 3 on hub 0:

```
hub ethernet 0 3
shutdown
```

SNMP Illegal Address Trap Enablement for Hub Port Example

The following example specifies the gateway IP address and enables an SNMP trap to be issued to the host 172.16.40.51 when a MAC address violation is detected on hub ports 2, 3, or 4. It specifies that Ethernet interface 0 is the source for all traps on the router. The community string is defined as the string *public* and the read/write parameter is set.

```
ip route 0.0.0.0 0.0.0.0 172.22.10.1
snmp-server community public rw
snmp-server trap-source ethernet 0
snmp-server host 172.16.40.51 public
hub ethernet 0 2 4
snmp trap illegal-address
```

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



Configuring Serial Interfaces

This module describes the procedure to configure serial interfaces. For hardware technical descriptions and information about installing interfaces, refer to the hardware installation and configuration publication for your product.

- [Finding Feature Information, page 59](#)
- [Prerequisites for Configuring Serial Interfaces, page 60](#)
- [Restrictions for Configuring Serial Interfaces, page 60](#)
- [Information About Configuring Serial Interfaces, page 60](#)
- [How to Configure Serial Interfaces, page 65](#)
- [Configuring a High-Speed Serial Interface, page 65](#)
- [Configuring a Synchronous Serial Interface, page 66](#)
- [Configuring a Channelized T3 Interface Processor, page 84](#)
- [Configuring PA-E3 and PA-2E3 Serial Port Adapters, page 100](#)
- [Configuring PA-T3 and PA-2T3 Serial Port Adapters, page 102](#)
- [Configuring a Packet OC-3 Interface, page 106](#)
- [Configuring a DPT OC-12c Interface, page 110](#)
- [Configuring Automatic Protection Switching of Packet-over-SONET Circuits, page 114](#)
- [Configuring Serial Interfaces for CSU DSU Service Modules, page 119](#)
- [Configuring Low-Speed Serial Interfaces, page 127](#)
- [Automatic Removal of tftp ftp rcp Source Interfaces Configuration, page 133](#)
- [Troubleshooting Serial Interfaces, page 134](#)
- [Configuration Examples for Serial Interface Configuration, page 142](#)
- [Additional References, page 160](#)
- [Feature Information for Configuring Serial Interfaces, page 161](#)

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.

Prerequisites for Configuring Serial Interfaces

The following are the prerequisites for configuring serial interfaces:

- Your hardware must support T3/E3 controllers and serial interfaces. The following hardware supports T3/E3 controllers and serial interfaces:
 - 2-Port and 4-Port Clear Channel T3/E3 SPAs
 - 2-Port and 4-Port Channelized T3 SPAs
- You have already configured the clear channel T3/E3 controller or channelized T3-to-T1/E1 controller that is associated with the serial interface you want to configure.

Restrictions for Configuring Serial Interfaces

In case of auto installation over a serial interface using either HDLC or Frame Relay, it can be performed only over the first serial port on a new device (serial interface 0 or serial interface x/0).

Information About Configuring Serial Interfaces

- [Cisco HDLC Encapsulation, page 60](#)
- [PPP Encapsulation, page 61](#)
- [Keepalive Timer, page 62](#)
- [Frame Relay Encapsulation, page 62](#)
- [Layer 2 Tunnel Protocol Version 3-Based Layer 2 VPN on Frame Relay, page 64](#)
- [High-Speed Serial Interfaces, page 64](#)

Cisco HDLC Encapsulation

Cisco High-Level Data Link Controller (HDLC) is the Cisco proprietary protocol for sending data over synchronous serial links using HDLC. Cisco HDLC also provides a simple control protocol called Serial Line Address Resolution Protocol (SLARP) to maintain serial link keepalives. Cisco HDLC is the default for data encapsulation at Layer 2 (data link) of the Open System Interconnection (OSI) stack for efficient packet delineation and error control.

**Note**

Cisco HDLC is the default encapsulation type for the serial interfaces.

When the encapsulation on a serial interface is changed from HDLC to any other encapsulation type, the configured serial subinterfaces on the main interface inherit the newly changed encapsulation and they do not get deleted.

Cisco HDLC uses keepalives to monitor the link state, as described in the [Keepalive Timer, page 62](#).

PPP Encapsulation

PPP is a standard protocol used to send data over synchronous serial links. PPP also provides a Link Control Protocol (LCP) for negotiating properties of the link. LCP uses echo requests and responses to monitor the continuing availability of the link.

**Note**

When an interface is configured with PPP encapsulation, a link is declared down, and full LCP negotiation is re-initiated after five ECHOREQ packets are sent without receiving an ECHOREP response.

PPP provides the following Network Control Protocols (NCPs) for negotiating properties of data protocols that will run on the link:

- IP Control Protocol (IPCP) to negotiate IP properties
- Multiprotocol Label Switching control processor (MPLSCP) to negotiate MPLS properties
- Cisco Discovery Protocol control processor (CDPCP) to negotiate CDP properties
- IPv6CP to negotiate IP Version 6 (IPv6) properties
- Open Systems Interconnection control processor (OSICP) to negotiate OSI properties

PPP uses keepalives to monitor the link state, as described in the [Keepalive Timer](#), page 62.

PPP supports the following authentication protocols, which require a remote device to prove its identity before allowing data traffic to flow over a connection:

- Challenge Handshake Authentication Protocol (CHAP)--CHAP authentication sends a challenge message to the remote device. The remote device encrypts the challenge value with a shared secret and returns the encrypted value and its name to the local router in a response message. The local router attempts to match the remote device's name with an associated secret stored in the local username or remote security server database; it uses the stored secret to encrypt the original challenge and verify that the encrypted values match.
- Microsoft Challenge Handshake Authentication Protocol (MS-CHAP)--MS-CHAP is the Microsoft version of CHAP. Like the standard version of CHAP, MS-CHAP is used for PPP authentication; in this case, authentication occurs between a personal computer using Microsoft Windows NT or Microsoft Windows 95 and a Cisco router or access server acting as a network access server.
- Password Authentication Protocol (PAP)--PAP authentication requires the remote device to send a name and a password, which are checked against a matching entry in the local username database or in the remote security server database.

Use the **ppp authentication** command in interface configuration mode to enable CHAP, MS-CHAP, and PAP on a serial interface.

**Note**

Enabling or disabling PPP authentication does not effect the local router's willingness to authenticate itself to the remote device.

- [Multilink PPP](#), page 61

Multilink PPP

Multilink Point-to-Point Protocol (MLPPP) is supported on the 1-Port Channelized OC-12/DS0 SPAs. MLPPP provides a method for combining multiple physical links into one logical link. The implementation

of MLPPP combines multiple PPP serial interfaces into one multilink interface. MLPPP performs the fragmenting, reassembling, and sequencing of datagrams across multiple PPP links. MLPPP is supported on the 2-Port and 4-Port Channelized T3 SPAs.

MLPPP provides the same features that are supported on PPP Serial interfaces with the exception of QoS. It also provides the following additional features:

- Fragment sizes of 128, 256, and 512 bytes
- Long sequence numbers (24-bit)
- Lost fragment detection timeout period of 80 ms
- Minimum-active-links configuration option
- LCP echo request/reply support over multilink interface
- Full T1 and E1 framed and unframed links

Keepalive Timer

Cisco keepalives are useful for monitoring the link state. Periodic keepalives are sent to and received from the peer at a frequency determined by the value of the keepalive timer. If an acceptable keepalive response is not received from the peer, the link makes the transition to the down state. As soon as an acceptable keepalive response is obtained from the peer or if keepalives are disabled, the link makes the transition to the up state.



Note

The **keepalive** command applies to serial interfaces using HDLC or PPP encapsulation. It does not apply to serial interfaces using Frame Relay encapsulation.

For each encapsulation type, a certain number of keepalives ignored by a peer triggers the serial interface to transition to the down state. For HDLC encapsulation, three ignored keepalives causes the interface to be brought down. For PPP encapsulation, five ignored keepalives causes the interface to be brought down. ECHOREQ packets are sent out only when LCP negotiation is complete (for example, when LCP is open).

Use the **keepalive** command in interface configuration mode to set the frequency at which LCP sends ECHOREQ packets to its peer. To restore the system to the default keepalive interval of 10 seconds, use the **keepalive** command with **no** argument. To disable keepalives, use the **keepalive disable** command. For both PPP and Cisco HDLC, a keepalive of 0 disables keepalives and is reported in the **show running-config** command output as **keepalive disable**.

When LCP is running on the peer and receives an ECHOREQ packet, it responds with an echo reply (ECHOREP) packet, regardless of whether keepalives are enabled on the peer.

Keepalives are independent between the two peers. One peer end can have keepalives enabled; the other end can have them disabled. Even if keepalives are disabled locally, LCP still responds with ECHOREP packets to the ECHOREQ packets it receives. Similarly, LCP also works if the period of keepalives at each end is different.

Frame Relay Encapsulation

When Frame Relay encapsulation is enabled on a serial interface, the interface configuration is hierarchical and comprises the following elements:

- The serial main interface comprises the physical interface and port. If you are not using the serial interface to support Cisco HDLC and PPP encapsulated connections, then you must configure subinterfaces with permanent virtual circuits (PVCs) under the serial main interface. Frame Relay connections are supported on PVCs only.

- Serial subinterfaces are configured under the serial main interface. A serial subinterface does not actively carry traffic until you configure a PVC under the serial subinterface. Layer 3 configuration typically takes place on the subinterface.
- When the encapsulation on a serial interface is changed from HDLC to any other encapsulation type, the configured serial subinterfaces on the main interface inherit the newly changed encapsulation and they do not get deleted.
- Point-to-point PVCs are configured under a serial subinterface. You cannot configure a PVC directly under a main interface. A single point-to-point PVC is allowed per subinterface. PVCs use a predefined circuit path and fail if the path is interrupted. PVCs remain active until the circuit is removed from either configuration. Connections on the serial PVC support Frame Relay encapsulation only.

**Note**

The administrative state of a parent interface drives the state of the subinterface and its PVC. When the administrative state of a parent interface or subinterface changes, so does the administrative state of any child PVC configured under that parent interface or subinterface.

To configure Frame Relay encapsulation on serial interfaces, use the **encapsulation (Frame Relay VC-bundle)** command.

Frame Relay interfaces support two types of encapsulated frames:

- Cisco (default)
- IETF

Use the **encap** command in PVC configuration mode to configure Cisco or IETF encapsulation on a PVC. If the encapsulation type is not configured explicitly for a PVC, then that PVC inherits the encapsulation type from the main serial interface.

**Note**

Cisco encapsulation is required on serial main interfaces that are configured for MPLS. IETF encapsulation is not supported for MPLS.

Before you configure Frame Relay encapsulation on an interface, you must verify that all prior Layer 3 configuration is removed from that interface. For example, you must ensure that there is no IP address configured directly under the main interface; otherwise, any Frame Relay configuration done under the main interface will not be viable.

- [LMI on Frame Relay Interfaces, page 63](#)

LMI on Frame Relay Interfaces

The Local Management Interface (LMI) protocol monitors the addition, deletion, and status of PVCs. LMI also verifies the integrity of the link that forms a Frame Relay UNI interface. By default, **cisco** LMI is enabled on all PVCs.

If the LMI type is **cisco** (the default LMI type), the maximum number of PVCs that can be supported under a single interface is related to the MTU size of the main interface. Use the following formula to calculate the maximum number of PVCs supported on a card or SPA:

$$(MTU - 13)/8 = \text{maximum number of PVCs}$$

**Note**

The default setting of the **mtu** command for a serial interface is 1504 bytes. Therefore, the default numbers of PVCs supported on a serial interface configured with **cisco** LMI is 186.

Layer 2 Tunnel Protocol Version 3-Based Layer 2 VPN on Frame Relay

The Layer 2 Tunnel Protocol Version 3 (L2TPv3) feature defines the L2TP protocol for tunneling Layer 2 payloads over an IP core network using Layer 2 virtual private networks (VPNs).

L2TPv3 is a tunneling protocol used for transporting Layer 2 protocols. It can operate in a number of different configurations and tunnel a number of different Layer 2 protocols and connections over a packet-switched network.

Before you can configure L2TPv3, you need configure a connection between the two attachment circuits (ACs) that will host the L2TPv3 pseudowire. This module describes how to configure a Layer 2 AC on a Frame Relay encapsulated serial interface.

**Note**

Serial interfaces support DLCI mode layer 2 ACs only; layer 2 port mode ACs are not supported on serial interfaces.

For detailed information about configuring L2TPv3 in your network, see the Cisco IOS Multiprotocol Label Switching Configuration Guide. For detailed information about configuring L2VPNs, see the L2VPN Interworking chapter in the Cisco IOS Multiprotocol Label Switching Configuration Guide.

High-Speed Serial Interfaces

The High-Speed Serial Interface (HSSI) Interface Processor (HIP) provides a single HSSI network interface. The network interface resides on a modular interface processor that provides a direct connection between the high-speed CiscoBus and an external network.

The HSSI port adapters (PA-H and PA-2H) are available on:

- Cisco 7200 series routers
- Second-generation Versatile Interface Processors (VIP2s) in Cisco 7500 series routers
- Cisco 7000 series routers with the 7000 series Route Switch Processor (RSP7000) and 7000 series Chassis Interface (RSP7000CI)

The PA-H provides one high-speed synchronous serial interface, and the PA-2H provides two high-speed synchronous serial interfaces that support full-duplex and data rates up to 52 Mbps. For more information on the PA-H, refer to the *PA-H HSSI Port Adapter Installation and Configuration* publication. For more information on the PA-2H, refer to the *PA-2H Dual-Port HSSI Port Adapter Installation and Configuration* publication.

The Cisco 3600 series 1-port HSSI network module provides full-duplex connectivity at SONET OC-1/STS-1 (51.840 MHz), T3 (44.736 MHz), and E3 (34.368 MHz) rates in conformance with the EIA/TIA-612 and EIA/TIA-613 specifications. The actual rate of the interface depends on the external data service unit (DSU) and the type of service to which it is connected. This 1-port HSSI network module can reach speeds of up to 52 Mbps in unidirectional traffic with 1548-byte packets and 4250 packets per second. ATM, High-Level Data Link Control (HDLC), PPP, Frame Relay, and Switched Multimegabit Data Service (SMDS) WAN services are all fully supported.

Before you configure the 1-port HSSI network module, complete the following prerequisite tasks:

- Install the HSSI Network Module in a chassis slot. For information on how to install this network module, refer to the "Installing a 1-Port HSSI Network Module in a Chassis Slot" section in the *1-Port HSSI Network Module Configuration Note* publication.
- Complete basic device configuration, including host name, user name, protocol, and security configuration. For more information about basic device configuration, refer to the *Cisco3620 Installation and Configuration Guide* or the *Cisco3640 Installation and Configuration Guide*.

How to Configure Serial Interfaces

Configuring a High-Speed Serial Interface

- [Specifying a HSSI Interface, page 65](#)
- [Specifying HSSI Encapsulation, page 65](#)
- [Invoking ATM on a HSSI Line, page 66](#)
- [Converting HSSI to Clock Master, page 66](#)

Specifying a HSSI Interface

To specify a High-Speed Serial Interface (HSSI) and enter interface configuration mode, use one of the following commands in global configuration mode.

Command	Purpose
Router(config)# interface hssi <i>number</i>	Enters interface configuration.
Router(config)# interface hssi <i>slot / port</i>	Enters interface configuration for the Cisco 7500 series routers.

Specifying HSSI Encapsulation

The HSSI supports the serial encapsulation methods, except for X.25-based encapsulations. The default method is HDLC. To define the encapsulation method, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# encapsulation { atm-dxi hdlc frame-relay ppp sdhc-primary sdhc-secondary smds }	Configures HSSI encapsulation.

For information about PPP, refer to the "Configuring Asynchronous SLIP and PPP" and "Configuring Media-Independent PPP and Multilink PPP" chapters in the *CiscoIOS Dial Technologies Configuration Guide*.

Invoking ATM on a HSSI Line

If you have an ATM DSU, you can invoke ATM over a HSSI line. You do so by mapping an ATM virtual path identifier (VPI) and virtual channel identifier (VCI) to a Data Exchange Interface (DXI) frame address. ATM-DXI encapsulation defines a data exchange interface that allows a DTE (such as a router) and a DCE (such as an ATM DSU) to cooperate to provide a User-Network Interface (UNI) for ATM networks.

To invoke ATM over a serial line, use the following commands in interface configuration mode.

SUMMARY STEPS

1. Router(config-if)# **encapsulation atm-dxi**
2. Router(config-if)# **dxi map protocol address vpi vci [broadcast]**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config-if)# encapsulation atm-dxi	Specifies the encapsulation method.
Step 2	Router(config-if)# dxi map protocol address vpi vci [broadcast]	Maps a given VPI and VCI to a DXI frame address.

You can also configure the **dxi map** command on a serial interface.

To configure an ATM interface using an ATM Interface Processor (AIP) card, refer to the "Configuring ATM" chapter in the *Cisco IOS Asynchronous Transfer Mode Configuration Guide*.

Converting HSSI to Clock Master

The HSSI network module provides full-duplex connectivity at SONET OC-1/STS-1 (51.840 MHz), T3 (44.736 MHz), and E3 (34.368 MHz) rates in conformance with the EIA/TIA-612 and EIA/TIA-613 specifications. The actual rate of the interface depends on the DSU and the type of service to which it is connected. To convert the HSSI interface into a clock master use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# hssi internal-clock	Converts the HSSI interface into a 51.84-MHz clock master.

Configuring a Synchronous Serial Interface

Synchronous serial interfaces are supported on various serial network interface cards or systems. These interfaces support full-duplex operation at T1 (1.544 Mbps) and E1 (2.048 Mbps) speeds. Refer to the *Cisco Product Catalog* for specific information regarding platform and hardware compatibility.

- [Synchronous Serial Configuration Task List, page 67](#)

Synchronous Serial Configuration Task List

To configure a synchronous serial interface, perform the tasks in the following sections. Each task in the list is identified as either required or optional.

See the [Configuration Examples for Serial Interface Configuration](#), page 142 for examples of configuration tasks described in this chapter.

- [Specifying a Synchronous Serial Interface](#), page 67
- [Specifying Synchronous Serial Encapsulation](#), page 68
- [Configuring PPP](#), page 69
- [Configuring Half-Duplex and Bisync for Synchronous Serial Port Adapters on Cisco 7200 Series Routers](#), page 69
- [Configuring Compression Service Adapters on Cisco 7500 Series Routers](#), page 70
- [Configuring Compression of HDLC Data](#), page 70
- [Configuring Real-Time Transport Protocol Header Compression](#), page 71
- [Configuring the Data Compression AIM](#), page 71
- [Configuring the CRC](#), page 79
- [Using the NRZI Line-Coding Format](#), page 80
- [Enabling the Internal Clock](#), page 80
- [Inverting the Data](#), page 80
- [Inverting the Transmit Clock Signal](#), page 81
- [Setting Transmit Delay](#), page 81
- [Configuring DTR Signal Pulsing](#), page 81
- [Ignoring DCD and Monitoring DSR as Line Up Down Indicator](#), page 82
- [Specifying the Serial Network Interface Module Timing](#), page 82
- [Specifying G.703 and E1-G.703 G.704 Interface Options](#), page 83

Specifying a Synchronous Serial Interface

To specify a synchronous serial interface and enter interface configuration mode, use one of the following commands in global configuration mode.

Command	Purpose
Router(config)# interface serial <i>number</i>	Enters interface configuration mode.
Router(config)# interface serial <i>slot / port</i>	Enters interface configuration mode for the Cisco 7200 or Cisco 7500 series routers.
Router(config)# interface serial <i>slot / port-adapter / port</i>	Enters interface configuration for the Cisco 7500 series routers.

Command	Purpose
<pre>Router(config)# interface serial slot / port:channel-group (Cisco 7000 series)</pre>	Enters interface configuration for a channelized T1 or E1 interface.
<pre>Router(config)# interface serial number:channel-group (Cisco 4000 series)</pre>	

Specifying Synchronous Serial Encapsulation

By default, synchronous serial lines use the High-Level Data Link Control (HDLC) serial encapsulation method, which provides the synchronous framing and error detection functions of HDLC without windowing or retransmission. The synchronous serial interfaces support the following serial encapsulation methods:

- ATM-DXI
- HDLC
- Frame Relay
- PPP
- Synchronous Data Link Control (SDLC)
- SMDS
- Cisco Serial Tunnel (STUN)
- X.25-based encapsulations

To define the encapsulation method, use the following command in interface configuration mode.

Command	Purpose
<pre>Router(config-if)# encapsulation {atm-dxi hdlc frame-relay ppp sdlc-primary sdlc-secondary smds stun x25}</pre>	Configures synchronous serial encapsulation.



Note

You cannot use the **physical-layer async** command for frame-relay encapsulation.

Encapsulation methods are set according to the type of protocol or application you configure in the Cisco IOS software.

- ATM-DXI is described in the [Configuring the CRC, page 79](#).
- PPP is described in the "Configuring Media-Independent PPP and Multilink PPP" chapter in the *Cisco IOS Dial Technologies Configuration Guide*.
- ATM, Frame Relay, and X.25 information and configuration steps are described in the *Cisco IOS Asynchronous Transfer Mode Configuration Guide* and the *Cisco IOS Wide-Area Networking Configuration Guide*.
- The remaining encapsulation methods are defined in their respective books and chapters describing the protocols or applications. Serial encapsulation methods are also discussed in the *Cisco IOS Interface and Hardware Component Command Reference*, under the **encapsulation** command.

By default, synchronous interfaces operate in full-duplex mode. To configure an SDLC interface for half-duplex mode, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# half-duplex</code>	Configures an SDLC interface for half-duplex mode.

Binary synchronous communication (Bisync) is a half-duplex protocol. Each block of transmission is acknowledged explicitly. To avoid the problem associated with simultaneous transmission, there is an implicit role of primary and secondary station. The primary sends the last block again if there is no response from the secondary within the period of block receive timeout.

To configure the serial interface for full-duplex mode, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# full-duplex</code>	Specifies that the interface can run Bisync using switched RTS signals.

Configuring PPP

To configure PPP, refer to the "Configuring Media-Independent PPP and Multilink PPP" chapter in the *CiscoIOS Dial Technologies Configuration Guide*.

Configuring Half-Duplex and Bisync for Synchronous Serial Port Adapters on Cisco 7200 Series Routers

The synchronous serial port adapters (PA-8T-V35, PA-8T-X21, PA-8T-232, and PA-4T+) on Cisco 7200 series routers support half-duplex and Bisync. Bisync is a character-oriented data-link layer protocol for half-duplex applications. In half-duplex mode, data is sent one direction at a time. Direction is controlled by handshaking the Request to Send (RST) and Clear to Send (CTS) control lines. These are described in the following sections:

For more information about the PA-8T-V35, PA-8T-X21, PA-8T-232, and PA-4T+ synchronous serial port adapters, refer to the following publications:

- *PA-8T-V35 Synchronous Serial Port Adapter Installation and Configuration*
- *PA-8T-X21 Synchronous Serial Port Adapter Installation and Configuration*
- *PA-8T-232 Synchronous Serial Port Adapter Installation and Configuration*
- *PA-4T+ Synchronous Serial Port Adapter Installation and Configuration*
- [Configuring Bisync, page 69](#)
- [Configuring Half-Duplex Carrier Modes and Timers, page 70](#)

Configuring Bisync

To configure the Bisync feature on the synchronous serial port adapters (PA-8T-V35, PA-8T-X21, PA-8T-232, and PA-4T+) on Cisco 7200 series routers, refer to the "Block Serial Tunnelling (BSTUN)" section of the "Configuring Serial Tunnel and Block Serial Tunnel" chapter of the *CiscoIOS Bridging and IBM Networking Configuration Guide*. All commands listed in the "Block Serial Tunnelling (BSTUN)"

Overview " section apply to the synchronous serial port adapters on Cisco 7200 series routers. Any command syntax that specifies an interface *number* supports the Cisco 7200 series *slot / port* syntax.

Configuring Half-Duplex Carrier Modes and Timers

Configuring Compression Service Adapters on Cisco 7500 Series Routers

The SA-Comp/1 and SA-Comp/4 data compression service adapters (CSAs) are available on:

- Cisco 7200 series routers
- Second-generation Versatile Interface Processors (VIP2s) in Cisco 7500 series routers (CSAs require VIP2 model VIP2-40.)

The SA-Comp/1 supports up to 64 WAN interfaces, and the SA-Comp/4 supports up to 256 WAN interfaces.

On the Cisco 7200 series routers you can optionally specify which CSA the interface uses to perform hardware compression.

You can configure point-to-point compression on serial interfaces that use PPP encapsulation. Compression reduces the size of a PPP frame via lossless data compression. PPP encapsulations support both predictor and Stacker compression algorithms.



Note

If the majority of your traffic is already compressed files, do not use compression.

When you configure Stacker compression on Cisco 7200 series routers and on Cisco 7500 series routers, there are three methods of compression: hardware compression, distributed compression, and software compression. Specifying the **compress stac** command with no options causes the router to use the fastest available compression method, as described here:

- If the router contains a compression service adapter (CSA), compression is performed in the CSA hardware (hardware compression).
- If the CSA is not available, compression is performed in the software installed on the VIP2 (distributed compression).
- If the VIP2 is not available, compression is performed in the router's main processor (software compression).

Using hardware compression in the CSA frees the main processor of the router for other tasks. You can also configure the router to use the VIP2 to perform compression by using the **distributed** option on the **compress** command, or to use the main processor of the router by using the **software** option on the **compress** command. If the VIP2 is not available, compression is performed in the main processor of the router.

When compression is performed in software installed in the main processor of the router, it might significantly affect system performance. You should disable compression in the router's main processor if the router CPU load exceeds 40 percent. To display the CPU load, use the **show process cpu EXEC** command.

For instructions on configuring compression over PPP, refer to the " Configuring Media-Independent PPP and Multilink PPP " chapter in the *CiscoIOS Dial Technologies Configuration Guide* .

Configuring Compression of HDLC Data

You can configure point-to-point software compression on serial interfaces that use HDLC encapsulation. Compression reduces the size of a HDLC frame via lossless data compression. The compression algorithm used is a Stacker (LZS) algorithm.

Compression is performed in software and might significantly affect system performance. We recommend that you disable compression if CPU load exceeds 65 percent. To display the CPU load, use the **show process cpu EXEC** command.

If the majority of your traffic is already compressed files, you should not use compression.

To configure compression over HDLC, use the following commands in interface configuration mode.

SUMMARY STEPS

1. Router(config-if)# **encapsulation hdlc**
2. Router(config-if)# **compress stac**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config-if)# encapsulation hdlc	Enables encapsulation of a single protocol on the serial line.
Step 2	Router(config-if)# compress stac	Enables compression.

Configuring Real-Time Transport Protocol Header Compression

Real-time Transport Protocol (RTP) is a protocol used for carrying packetized audio and video traffic over an IP network. RTP is described in RFC 1889, *>RTP--A Transport Protocol for Real-Time Applications*. RTP is not intended for data traffic, which uses TCP or UDP (User Datagram Protocol). RTP provides end-to-end network transport functions intended for applications with real-time requirements, such as audio, video, or simulation data over multicast or unicast network services.

For information and instructions for configuring RTP header compression, refer to the *Cisco IOS IP Multicast Configuration Guide*.

Configuring the Data Compression AIM

The data compression Advanced Interface Module (AIM) provides hardware-based compression and decompression of packet data transmitted and received on the serial network interfaces of the Cisco 2600 series router without occupying the Port Module Slot which might otherwise be used for additional customer network ports. Supported are the industry standard Lempel-Ziv Stac (LZS) and Microsoft point-to-point compression (MPPC) compression algorithms over point-to-point protocol (PPP) or Frame Relay. High-level Data Link Control (HDLC) is not supported. The data compression AIM requires Cisco IOS Release 12.0(1)T or later.

The data compression AIM is a daughtercard assembly that attaches directly to the Cisco 2600 motherboard leaving the single network module slot available for other purposes. The data compression AIM supports only serial interfaces using PPP encapsulation with STAC or MPPC compression, or Frame Relay encapsulation with STAC compression. No routing, bridging, or switching performance is impacted by this feature. The data compression AIM module contains a high-performance data compression coprocessor that implements the LZS and MPPC data compression algorithms. The module provides compression support for up to two E1 lines. The module contains a PCI Target/Initiator system bus interface for access into host system memory with minimal Host processor intervention.

To configure the data compression AIM daughtercard assembly, perform the following tasks:

- [Configuring PPP Compression, page 72](#)
- [Verifying PPP Compression, page 73](#)
- [Configuring Frame Relay Map Compression, page 73](#)
- [Verifying Frame Relay Map Compression, page 74](#)
- [Configuring Frame Relay Payload Compression, page 76](#)
- [Verifying Frame Relay Payload Compression, page 77](#)
- [Configuring Diagnostics, page 78](#)
- [Verifying Diagnostics, page 78](#)

Configuring PPP Compression

Configure your Cisco 2600 access server to use PPP compression. Specify the following information for each serial interface:

- encapsulation type
- compression algorithm
- the CAIM daughtercard to be designated as the source of this algorithm, and the port.

To configure the PPP form of compression, use the following commands, beginning in privileged EXEC mode.

SUMMARY STEPS

1. Router# **configure terminal**
2. Router(config)# **interface serial slot / port**
3. Router(config-if)# **encapsulation ppp**
4. Router(config-if)# **compress{mppcstac} caim element-number**
5. Router(config-if)# **no shutdown**
6. Router(config-if)# **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# interface serial slot / port	Enters interface configuration mode to configure serial interface 0 on port 0. If you have installed more than one WAN interface card, you have interfaces 0 and 1. Each WAN interface card has a pair of ports, 0 and 1.
Step 3	Router(config-if)# encapsulation ppp	Specifies the ppp encapsulation type. ¹¹
Step 4	Router(config-if)# compress{mppcstac} caim element-number	Specifies one of the algorithms (mppc, predictor, or stac) on the CAIM card for port 0. ¹²
Step 5	Router(config-if)# no shutdown	Restarts the interface.
Step 6	Router(config-if)# exit	Returns to EXEC mode.

¹¹ You also have the option of configuring encapsulation for Frame Relay.

¹² You can also configure compression for another serial port or another CAIM card, depending upon your configuration.

Verifying PPP Compression

To check that the interface is activated, use the **show interfaces serial slot / port** command. Notice the highlighted fields in the following example:

```
Router# show interfaces serial 0/0
Serial0/0 is up, line protocol is up
  Hardware is PowerQUICC Serial
  Internet address is 1.1.1.2/24
  MTU 1500 bytes, BW 2000 Kbit, DLY 20000 usec,
    reliability 255/255, txload 3/255, rxload 50/255
  Encapsulation PPP
, loopback not set, keepalive not set
  LCP Open
  Open: IPCP, CCP ==> If two routers have successfully negotiated compression.
  Last input 00:00:04, output 00:00:00, output hang never
  Last clearing of "show interface" counters lwld
  Queueing strategy: fifo
  Output queue
0/40, 80 drops; input queue
0/75, 0 drops
  30 second input rate 397000 bits/sec, 40 packets/sec
  30 second output rate 30000 bits/sec, 40 packets/sec
  27859655 packets input, 4176659739 bytes, 0 no buffer
  Received 175145 broadcasts, 0 runts, 0 giants, 0 throttles
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  55309592 packets output, 1044865717 bytes, 0 underruns
  0 output errors, 0 collisions, 12 interface resets
  0 output buffer failures, 0 output buffers swapped out
  36 carrier transitions
  DCD=up DSR=up DTR=up RTS=up CTS=up
```

To indicate whether compression is active, use the **show compress** command. Notice the highlighted fields in the following example:

```
Router# show compress
Serial0/0
  Hardware compression enabled
  CSA in slot 0 in use
  Compressed bytes sent:
317862131 bytes   61 Kbits/sec  ratio: 12.870
  Compressed bytes rcv:
221975672 bytes  43 Kbits/sec  ratio: 9.194
  restarts: 1
  last clearing of counters: 41252 seconds
```



Tip

The interface must report being up.

- No errors should be reported.
- Check this interface again after you are sure that traffic is getting to the Cisco 2600 series router and verify that the **Compressed bytes rcv** field value changes.

Configuring Frame Relay Map Compression

Configure Frame Relay to map compression on this Data-Link Connection Identifier (DLCI) to use the specified AIM hardware compression on the Cisco 2600 access server. You must specify the following information for each serial interface:

- The protocol, protocol address
- DLCI
- Encapsulation type

- FRF.9 stac compression algorithm

You must also designate the CAIM daughtercard as a source of this algorithm, and the CAIM element number.

To configure the Frame Relay map compression command for operation, use the following commands beginning in privileged EXEC mode.

SUMMARY STEPS

1. Router# **configure terminal**
2. Router(config)# **interface serial slot / port**
3. Router(config-if)# **encapsulation frame-relay**
4. Router(config-if)# **frame-relay map ip ip-address dlc-number broadcast payload-compression frf9 stac caim element-number**
5. Router(config-controller)# **no shutdown**
6. Router(config-if)# **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# interface serial slot / port	Enters interface configuration mode to configure the serial interface. If you have installed more than one WAN interface card, you have interfaces 0 and 1. Each WAN interface card has a pair of ports, 0 and 1.
Step 3	Router(config-if)# encapsulation frame-relay	Specifies Frame Relay encapsulation. ¹³
Step 4	Router(config-if)# frame-relay map ip ip-address dlc-number broadcast payload-compression frf9 stac caim element-number	Specifies the stac algorithm on the CAIM card for the port. ¹⁴
Step 5	Router(config-controller)# no shutdown	Restarts the interface.
Step 6	Router(config-if)# exit	Returns to EXEC mode.



Note

The **compress ppp** command applied to the PPP compression configuration example above has no equivalent for compression under Frame Relay.

Verifying Frame Relay Map Compression

To check that the interface is activated with proper compression and encapsulation, use the **show interfaces serial slot / port** command. Notice the highlighted fields in the following example:

```
Router# show interfaces serial 0/1
Serial0/1 is up, line protocol is up
```

¹³ You also have the option of configuring encapsulation for PPP.

¹⁴ You can also configure compression for another serial port or another CAIM card, depending upon your configuration.

```

Hardware is PowerQUICC Serial
Internet address is 1.1.1.2/24
MTU 1500 bytes, BW 2000 Kbit, DLY 20000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
Encapsulation FRAME-RELAY
, loopback not set, keepalive not set
FR SVC disabled, LAPF state down
Broadcast queue 0/64, broadcasts sent/dropped 2743/0, interface broadcasts 2742
Last input 03:05:57, output 00:00:03, output hang never
Last clearing of "show interface" counters lwld
Queueing strategy: fifo
Output queue
0/40, 80 drops; input queue
0/75, 0 drops
30 second input rate 0 bits/sec, 0 packets/sec
30 second output rate 0 bits/sec, 0 packets/sec
30800054 packets input, 3488155802 bytes, 0 no buffer
Received 199567 broadcasts, 0 runts, 0 giants, 0 throttles
2 input errors, 0 CRC, 2 frame, 0 overrun, 0 ignored, 0 abort
58246738 packets output, 1325052697 bytes, 0 underruns
0 output errors, 0 collisions, 15 interface resets
0 output buffer failures, 0 output buffers swapped out
36 carrier transitions
DCD=up DSR=up DTR=up RTS=up CTS=up

```

To indicate whether compression is active, use the **show controllers serial slot / port** command. Notice the highlighted fields in the following example:

```

Router# show controllers serial 1/0
CD2430 Slot 1, Port 0, Controller 0, Channel 0, Revision 14
Channel mode is synchronous serial
idb 0x811082E8, buffer size 1524, X.21 DTE cable
Global registers
    rpilr 0x2, rir 0x0, risr 0x0, rfoc 0x0, rdr 0x30
    tpilr 0x1, tir 0x0, tisir 0x60, tftc 0x0, tdr 0x41
    mpilr 0x3, mir 0x2, misr 0x60
    bercnt 0xFF, stk 0x0
Per-channel registers for channel 0
Option registers
0x02 0x00 0x42 0xE7 0xE0 0x00 0x00
Command and status registers
cmr 0xC0, ccr 0x00, csr 0xAC, msvr-rts 0xF1, msvr-dtr 0xF1
Clock option registers
rcor 0x06, rbpr 0x01, tcor 0xC8, tbpr 0x01
Interrupt registers
ier 0x89, livr 0x00, licr 0x00
DMA buffer status 0x27
DMA receive registers
arbaddr 0x2549D44, arbcnt 1548, arbsts 0x1
brbaddr 0x2548344, brbcnt 1548, brbsts 0x1
rcbaddr 0x2549D94
DMA transmit registers
atbaddr 0x257F93E, atbcnt 104, atbsts 0x43
btbaddr 0x25B25C2, btbcnt 1490, btbsts 0x43
tcbaddr 0x25B25D2
Special character registers
schr1 0x00, schr2 0x00, schr3 0x00, schr4 0x00
scrl 0x0, scrh 0x0, lnxt 0xF1
Driver context information
Context structure 0x8110D830, Register table 0x40800400
Serial Interface Control 5:1 Register (0x40800802) is 0x0
Adaptor Flags 0x0
Serial Modem Control Register (0x40800804) is 0x18
Receive static buffer 0x810E1274
Receive particle buffers 0x8110DE00, 0x8110DDC0
Transmit DMA buffers 0x8113E240, 0x810F2808, 0x810D4C00, 0x810EA0DC
Transmit packet with particles 0x0, first word is 0x0
Interrupt rates (per second) transmit 25, receive 139, modem 0
True fast-switched packets      41
Semi fast-switched packets     13449573
Transmitter hang count          0
Residual indication count      0
Bus error count                 0

```

```

Aborted short frames count    0
CRC short frames count       0
Error counters
  CTS deassertion failures    0
  Nested interrupt errors transmit 0, receive 0, modem 0
Using Compression AIM 0
CompressionAim0
  ds:0x8113FC04 idb:0x8113A6CC
    5005867 uncomp paks in --> 5005867 comp paks out
    38397501 comp paks in --> 38397502 uncomp paks out
    2882277146 uncomp bytes in--> 497476655 comp bytes out
    3500965085 comp bytes in --> 1211331227 uncomp bytes out
      72 uncomp paks/sec in--> 72 comp paks/sec out
      557 comp paks/sec in --> 557 uncomp paks/sec out
      334959 uncomp bits/sec in--> 57812 comp bits/sec out
      406855 comp bits/sec in --> 140827 uncomp bits/sec out
    68841 seconds since last clear
    holdq:0 hw_enable:1 src_limited:0 num cnxts:8
    no data:0 drops:0 nobuffers:0 enc adj errs:0 fallbacks:
5322165
  no Replace:0 num seq errs:0 num desc errs:0 cmds complete:
43403738
  Bad reqs:0 Dead cnxts:0 No Paks:0 enq errs:0
  rx pkt drops:0 tx pkt drops:0 dequeues:0 requeues:0
  drops disabled:0 clears:0 ints:41973007 purges:203200
  no cnxts:0 bad algos:0 no crams:0 bad paks:0
  # opens:0 # closes:4 # hangs:0
  # 9711 fatal:0 # poison pkts:0 cmd/res ovrns:0
  # dma fatal:0
  Jupiter DMA Controller Registers:(0x40200000
    Cmd Ring:0x025BAE60 Src Ring:0x025BBB60
    Res Ring:0x025BB4E8 Dst Ring:0x025BBDA8
    Status/Cntl:present:0x8080989C last int:0x9898989C
    Inten:0x30302021 config:0x00080003
    Num DMA ints:41973355
  Hifn9711 Data Compression Coprocessor Registers (0x40201000):
    Config:0x000051D4 Inten:0x00000E00
    Status:0x00004000 FIFO status:0x00004000
    FIFO config:0x00000101

```

**Tip**

The interface must report being up.

- No errors should be reported.
- Check this interface again after you are sure that traffic is getting to the Cisco 2600 series router and verify that the **Compressed bytes rcv** field value changes.

Configuring Frame Relay Payload Compression

To configure Frame Relay payload compression, use the following commands beginning in privileged EXEC mode.

SUMMARY STEPS

1. Router# **configure terminal**
2. Router(config)# **interface serial slot / port**
3. Router(config-if)# **encapsulation ppp**
4. Router(config-if)# **frame-relay payload-compression frf9 stac**
5. Router(config-if)# **no shutdown**
6. Router(config-if)# **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# interface serial slot / port	Enters interface configuration mode to configure the specified serial interface and port.
Step 3	Router(config-if)# encapsulation ppp	Specifies PPP encapsulation. ¹⁵
Step 4	Router(config-if)# frame-relay payload-compression frf9 stac	Specifies the stac algorithm on the CAIM card for the specified port. ¹⁶
	Example: <pre> caim element-number </pre>	
Step 5	Router(config-if)# no shutdown	Restarts the interface.
Step 6	Router(config-if)# exit	Returns to EXEC mode.

Verifying Frame Relay Payload Compression

To check that the interface is activated with proper compression and encapsulation, use the **show interfaces serial slot / port** command. Notice the highlighted fields in the following example:

```

Router# show interfaces serial 0/0
Serial0/0 is up, line protocol is up
  Hardware is PowerQUICC Serial
  Internet address is 1.1.1.2/24
  MTU 1500 bytes, BW 2000 Kbit, DLY 20000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation FRAME-RELAY,
  loopback not set, keepalive not set
  FR SVC disabled, LAPF state down
  Broadcast queue 0/64, broadcasts sent/dropped 2743/0, interface broadcasts 2742
  Last input 03:05:57, output 00:00:03, output hang never
  Last clearing of "show interface" counters 1w1d
  Queueing strategy: fifo
  Output queue
  0/40, 80 drops; input queue
  0/75, 0 drops
  30 second input rate 0 bits/sec, 0 packets/sec
  30 second output rate 0 bits/sec, 0 packets/sec
  30800054 packets input, 3488155802 bytes, 0 no buffer
  Received 199567 broadcasts, 0 runts, 0 giants, 0 throttles
  2 input errors, 0 CRC, 2 frame, 0 overrun, 0 ignored, 0 abort
  58246738 packets output, 1325052697 bytes, 0 underruns
  0 output errors, 0 collisions, 15 interface resets
  0 output buffer failures, 0 output buffers swapped out

```

¹⁵ You also have the option of configuring encapsulation for Frame Relay.

¹⁶ You can configure compression for any serial port or another CAIM card, depending upon your configuration.

```
36 carrier transitions
DCD=up DSR=up DTR=up RTS=up CTS=up
```

**Note**

FRAME-RELAY is not displayed using the **show compress** command. Use the **debug compress** command to see this information.

**Tip**

The interface must report being up.

- No errors should be reported.

Configuring Diagnostics

Configure the AIM daughtercard to provide compression for the Cisco 2600 series router. You must specify the following information for each daughtercard installed.

To configure the PPP for compression, use the following commands beginning in user EXEC mode.

SUMMARY STEPS

1. Router> **enable**
2. Router# **show pas caim stats** *element-number*
3. Router# **show compress**
4. Router# **clear compress**
5. Router# **show pas caim stats** *element-number*
6. Router# **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router> enable	Enables higher privilege levels, such as privileged EXEC mode.
Step 2	Router# show pas caim stats <i>element-number</i>	Displays compression statistics for your CAIM.
Step 3	Router# show compress	Displays the current configuration for compression on your Cisco 2600.
Step 4	Router# clear compress	Clears all the counters and resets the CAIM hardware.
Step 5	Router# show pas caim stats <i>element-number</i>	Displays compression statistics for your CAIM.
Step 6	Router# exit	Returns to EXEC mode.

Verifying Diagnostics

To check that the data compression AIM is collecting statistics that represent proper compression, use the **show pas caim stats** *element-number* command:

```
Router# show pas caim stats 0
CompressionAim0
  ds:0x80F56A44 idb:0x80F50DB8
    422074 uncompress paks in -->      422076 compress paks out
```

```

422071 comp paks in --> 422075 uncomp paks out
633912308 uncomp bytes in--> 22791798 comp bytes out
27433911 comp bytes in --> 633911762 uncomp bytes out
974 uncomp paks/sec in--> 974 comp paks/sec out
974 comp paks/sec in --> 974 uncomp paks/sec out
11739116 uncomp bits/sec in--> 422070 comp bits/sec out
508035 comp bits/sec in --> 11739106 uncomp bits/sec out
433 seconds since last clear
holdq: 0 hw_enable: 1 src_limited: 0 num cnxts: 4
no data: 0 drops: 0 nobuffers: 0 enc adj errs: 0 fallbacks: 0
no Replace: 0 num seq errs: 0 num desc errs: 0 cmds complete: 844151
Bad reqs: 0 Dead cnxts: 0 No Paks: 0 enq errs: 0
rx pkt drops: 0 tx pkt drops: 0 dequeues: 0 requeues: 0
drops disabled: 0 clears: 0 ints: 844314 purges: 0
no cnxts: 0 bad algos: 0 no crams: 0 bad paks: 0
# opens: 0 # closes: 0 # hangs: 0

```

To identify compression characteristics for each port, use the **show compress** command:

```

Router# show compress
Serial0/0
Hardware compression enabled
CSA in slot 0 in use
Compressed bytes sent: 317862131 bytes 61 Kbits/sec ratio: 12.870
Compressed bytes rcv: 221975672 bytes 43 Kbits/sec ratio: 9.194
restarts: 1
last clearing of counters: 41252 seconds
Serial0/1
Hardware compression enabled
CSA in slot 0 in use
Compressed bytes sent: 249720 bytes 0 Kbits/sec ratio: 5.923
Compressed bytes rcv: 465843659 bytes 43 Kbits/sec ratio: 9.128
restarts: 1
last clearing of counters: 85525 seconds

```

To reset the CAIM hardware to 0, use the **clear compress** command. There is no output for this command; instead, check the output from the **show compress** command to verify the result:

```

Router# clear compress
Router# show compress
Serial0/0
Hardware compression enabled
CSA in slot 0 in use
Compressed bytes sent: 0 bytes 61 Kbits/sec ratio: 0
Compressed bytes rcv: 0 bytes 43 Kbits/sec ratio: 0
restarts: 0
last clearing of counters: 0 seconds

```



Tip The interface must report being up.

- No errors should be reported.

Configuring the CRC

The cyclic redundancy check (CRC) on a serial interface defaults to a length of 16 bits. To change the length of the CRC to 32 bits on an Fast Serial Interface Processor (FSIP) or HSSI Interface Processor (HIP) of the Cisco 7500 series only, use the following command in interface configuration mode.

Command	Purpose
Router (config-if)# crc <i>size</i>	Sets the length of the CRC.

Using the NRZI Line-Coding Format

The nonreturn-to-zero (NRZ) and nonreturn-to-zero inverted (NRZI) formats are supported on:

- All FSIP interface types on Cisco 7500 series routers
- PA-8T and PA-4T+ synchronous serial port adapters on:
 - Cisco 7000 series routers with RSP7000
 - Cisco 7200 series routers
 - Cisco 7500 series routers

NRZ and NRZI are line-coding formats that are required for serial connections in some environments. NRZ encoding is most common. NRZI encoding is used primarily with EIA/TIA-232 connections in IBM environments.

The default configuration for all serial interfaces is NRZ format. The default is **no nrzi-encoding**.

To enable NRZI format, use one of the following commands in interface configuration mode.

Command	Purpose
Router(config-if)# nrzi-encoding	Enables NRZI encoding format.
or	
Router(config-if)# nrzi-encoding [mark]	Enables NRZI encoding format for Cisco 7200 series routers and Cisco 7500 series routers.

Enabling the Internal Clock

When a DTE does not return a transmit clock, use the following interface configuration command on the Cisco 7000 series to enable the internally generated clock on a serial interface:

Command	Purpose
Router(config-if)# transmit-clock-internal	Enables the internally generated clock on a serial interface.

Inverting the Data

If the interface on the PA-8T and PA-4T+ synchronous serial port adapters is used to drive a dedicated T1 line that does not have B8ZS encoding, you must invert the data stream on the connecting CSU/DSU or on the interface. Be careful not to invert data on both the CSU/DSU and the interface because two data inversions will cancel each other out.

If the T1 channel on the CT3IP is using alternate mark inversion (AMI) line coding, you must invert the data. For more information, refer to the **t1 linecode** controller configuration command. For more information on the CT3IP, see the [Configuring a Channelized T3 Interface Processor](#), page 84.

To invert the data stream, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# invert data</code>	Inverts the data on an interface.

Inverting the Transmit Clock Signal

Systems that use long cables or cables that are not transmitting the TxC signal (transmit echoed clock line, also known as TXCE or SCTE clock) can experience high error rates when operating at the higher transmission speeds. For example, if the interface on the PA-8T and PA-4T+ synchronous serial port adapters is reporting a high number of error packets, a phase shift might be the problem. Inverting the clock signal can correct this shift. To invert the clock signal, use the following commands in interface configuration mode.

Command	Purpose
<code>Router(config-if)# invert txclock</code>	Inverts the clock signal on an interface.
<code>Router(config-if)# invert rxclock</code>	Inverts the phase of the RX clock on the UIO serial interface, which does not use the T1/E1 interface.

Setting Transmit Delay

It is possible to send back-to-back data packets over serial interfaces faster than some hosts can receive them. You can specify a minimum dead time after transmitting a packet to remove this condition. This setting is available for serial interfaces on the MCI and SCI interface cards and for the HSSI or MIP. Use one of the following commands, as appropriate for your system, in interface configuration mode.

Command	Purpose
<code>Router(config-if)# transmitter-delay microseconds</code>	Sets the transmit delay on the MCI and SCI synchronous serial interfaces.
<code>Router(config-if)# transmitter-delay hdlc- flags</code>	Sets the transmit delay on the HSSI or MIP.

Configuring DTR Signal Pulsing

You can configure pulsing dedicated Token Ring (DTR) signals on all serial interfaces. When the serial line protocol goes down (for example, because of loss of synchronization), the interface hardware is reset and the DTR signal is held inactive for at least the specified interval. This function is useful for handling encrypting or other similar devices that use the toggling of the DTR signal to reset synchronization. To configure DTR signal pulsing, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# pulse-time seconds</code>	Configures DTR signal pulsing.

Ignoring DCD and Monitoring DSR as Line Up Down Indicator

This task applies to:

- Quad Serial NIM (network interface module) interfaces on the Cisco 4000 series
- Hitachi-based serial interfaces on the Cisco 2500 series and Cisco 3000 series

By default, when the serial interface is operating in DTE mode, it monitors the Data Carrier Detect (DCD) signal as the line up/down indicator. By default, the attached DCE device sends the DCD signal. When the DTE interface detects the DCD signal, it changes the state of the interface to up.

In some configurations, such as an SDLC multidrop environment, the DCE device sends the Data Set Ready (DSR) signal instead of the DCD signal, which prevents the interface from coming up. To tell the interface to monitor the DSR signal instead of the DCD signal as the line up/down indicator, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# ignore-dcd</code>	Configures the serial interface to monitor the DSR signal as the line up/down indicator.



Caution

Unless you know for certain that you really need this feature, be very careful using this command. It will hide the real status of the interface. The interface could actually be down and you will not know by looking at show displays.

Specifying the Serial Network Interface Module Timing

On Cisco 4000 series routers, you can specify the serial Network Interface Module timing signal configuration. When the board is operating as a DCE and the DTE provides terminal timing (SCTE or TT), you can configure the DCE to use SCTE from the DTE. When running the line at high speeds and long distances, this strategy prevents phase shifting of the data with respect to the clock.

To configure the DCE to use SCTE from the DTE, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# dce-terminal-timing enable</code>	Configures the DCE to use SCTE from the DTE.

When the board is operating as a DTE, you can invert the TXC clock signal it gets from the DCE that the DTE uses to transmit data. Invert the clock signal if the DCE cannot receive SCTE from the DTE, the data is running at high speeds, and the transmission line is long. Again, this prevents phase shifting of the data with respect to the clock.

To configure the interface so that the router inverts the TXC clock signal, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# dte-invert-txc	Specifies timing configuration to invert TXC clock signal.

Specifying G.703 and E1-G.703 G.704 Interface Options

This section describes the optional tasks for configuring a G.703 serial interface (a serial interface that meets the G.703 electrical and mechanical specifications and operates at E1 data rates). G.703 interfaces are available on port adapters for the Fast Serial Interface Processor (FSIP) on a Cisco 4000 series or Cisco 7500 series router.

The E1-G.703/G.704 serial port adapters (PA-4E1G-120 and PA-4E1G-75) are available on:

- Cisco 7500 series routers
- Cisco 7200 series routers
- Cisco 7000 series routers with the 7000 series Route Switch Processor (RSP7000) and 7000 series Chassis Interface (RSP7000CI)

These port adapters provide up to four E1 synchronous serial interfaces, which are compatible with and specified by G.703/G.704. The PA-4E1G-120 supports balanced operation, and the PA-4E1G-75 supports unbalanced operation with 15-pin, D-shell (DB-15) receptacles on the port adapters. Both port adapters operate in full-duplex mode at the E1 speed of 2.048 Mbps.

Configuration tasks are described in the following sections:

- [Enabling Framed Mode, page 83](#)
- [Enabling CRC4 Generation, page 83](#)
- [Using Time Slot 16 for Data, page 84](#)
- [Specifying a Clock Source, page 84](#)

Enabling Framed Mode

G.703 interfaces have two modes of operation: framed and unframed. By default, G.703 serial interfaces are configured for unframed mode. To enable framed mode, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# timeslot <i>start-slot</i> -- <i>stop-slot</i>	Enables framed mode.

To restore the default, use the **no** form of this command or set the starting time slot to 0.

Enabling CRC4 Generation

By default, the G.703 CRC4, which is useful for checking data integrity while operating in framed mode, is not generated. To enable generation of the G.703 CRC4, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# crc4	Enables CRC4 generation.

Using Time Slot 16 for Data

By default, time slot 16 is used for signaling. It can also be used for data (in order to get all possible subframes or time slots when in framed mode). To specify the use of time slot 16 for data, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# ts16	Specifies that time slot 16 is used for data.

Specifying a Clock Source

A G.703 interface can clock its transmitted data either from its internal clock or from a clock recovered from the receive data stream of the line. By default, the interface uses the receive data stream of the line. To control which clock is used, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# clock source { line internal loop-timed }	Specifies the clock used for transmitted data.

Configuring a Channelized T3 Interface Processor

The Channelized T3 Interface Processor (CT3IP) is available on:

- Cisco 7500 series routers
- Cisco 7000 series routers with the 7000 series Route Switch Processor (RSP7000) and 7000 series Chassis Interface (RSP7000CI)

The Channelized T3 (CT3) feature board is available on Cisco AS5800 access servers.

These cards provide for the aggregation of channelized interfaces into a single T3 facility. T3 support on the Cisco AS5800 allows support for 28 T1s (672 channels) per chassis. The Channelized T3 dual-wide port adapter (PA-CT3/4T1) can be used in Cisco 7200 series routers.



Note

Throughout this document are references to the CT3IP. However, the term CT3IP also applies to the PA-CT3/4T1 and to the CT3 feature board. Wherever you see a description of a feature of the CT3IP, the feature is also available in the PA-CT3/4T1 and the CT3 feature board, unless otherwise indicated.

The CT3IP is a fixed-configuration interface processor based on the second-generation Versatile Interface Processor (VIP2). The CT3 channelized port adapter (PA-CT3/4T1) is a dual-wide port adapter. The CT3IP or PA-CT3/4T1 has four T1 connections via DB-15 connectors and one DS3 connection via BNC connectors. Each DS3 interface can provide up to 28 T1 channels (a single T3 group). Each channel is presented to the system as a serial interface that can be configured individually. The CT3IP or PA-CT3/4T1

can transmit and receive data bidirectionally at the T1 rate of 1.536 Mbps. The four T1 connections use 100-ohm twisted-pair serial cables to external channel service units (CSUs) or to a MultiChannel Interface Processor (MIP) on the same router or on another router. For wide-area networking, the CT3IP or PA-CT3/4T1 can function as a concentrator for a remote site.

**Note**

The VIP2-50 is the newest and fastest second-generation Versatile Interface Processor (VIP2) available on Cisco 7500 series routers that use the Route Switch Processor (RSP), and on Cisco 7000 series routers with the 7000 Series Route Switch Processor (RSP7000) and 7000 Series Chassis Interface (RSP7000CI). The VIP2-50 provides significantly greater packet and program memory space and increased distributed switching performance. For more information on the VIP2-50, refer to the *Second-Generation Versatile Interface Processor (VIP2) Installation, Configuration, and Maintenance* publication.

As mentioned above, the CT3IP or PA-CT3/4T1 provides 28 T1 channels for serial transmission of data. Each T1 channel can be configured to use a portion of the T1 bandwidth or the entire T1 bandwidth for data transmission. Bandwidth for each T1 channel can be configured for $n \times 56$ kbps or $n \times 64$ kbps (where n is 1 to 24). The unused portion of the T1 bandwidth, when not running at full T1 speeds, is filled with idle channel data. The CT3IP or PA-CT3/4T1 does not support the aggregation of multiple T1 channels (called *inversemultiplexing* or *bonding*) for higher bandwidth data rates.

The first three T1 channels of the CT3IP or PA-CT3/4T1 can be broken out to the three DSUP-15 connectors on the CPT3IP or PA-CT3/4T1 so that the T1 can be further demultiplexed by the MIP on the same router or on another router or by other multiplexing equipment. When connecting to the MIP, you configure a channelized T1 as described in the [Configuring External T1 Channels, page 91](#). This is referred to as an external T1 channel.

The CT3IP supports RFC 1406, *Definitions of Managed Objects for DS1 and E1 Interface Types*, and RFC 1407, *DS3 MIB Variables* (CISCO-RFC-1407-CAPABILITY.my). For information about Cisco MIBs, refer to the current Cisco IOS release note for the location of the MIB online reference.

For RFC 1406, Cisco supports all tables except the "Frac" table. For RFC 1407, Cisco supports all tables except the "FarEnd" tables.

The CT3IP supports the following WAN protocols:

- Frame Relay
- HDLC
- PPP
- SMDS Data Exchange Interface (DXI)

The CT3IP meets ANSI T1.102-1987 and BELCORE TR-TSY-000499 specifications for T3 and meets ANSI 62411 and BELCORE TR499 specifications for T1. The CT3IP provides internal CSU functionality and includes reporting performance data statistics, transmit and receive statistics, and error statistics. The CT3IP supports RFC 1406 (T1 MIB) and RFC 1407 (T3 MIB).

External T1 channels do not provide CSU functionality and must connect to an external CSU.

- [Channelized T3 Configuration Task List, page 85](#)

Channelized T3 Configuration Task List

To configure the CT3IP, perform the tasks in the following sections. Each task is identified as either required or optional.

After you configure the T1 channels on the CT3IP, you can continue configuring it as you would a normal serial interface.

For CT3IP configuration examples, see the [Channelized T3 Interface Processor Configuration Examples](#), page 143.

- [Configuring T3 Controller Support for the Cisco AS5800](#), page 86
- [Configuring the T3 Controller](#), page 88
- [Configuring Each T1 Channel](#), page 89
- [Configuring External T1 Channels](#), page 91
- [Monitoring and Maintaining the CT3IP](#), page 92
- [Verifying T3 Configuration](#), page 93
- [Configuring Maintenance Data Link Messages](#), page 94
- [Enabling Performance Report Monitoring](#), page 94
- [Configuring for BERT on the Cisco AS5300](#), page 95
- [Verifying BERT on the Cisco AS5300](#), page 96
- [Enabling a BERT Test Pattern](#), page 96
- [Enabling Remote FDL Loopbacks](#), page 97
- [Configuring T1 Cable Length and T1 E1 Line Termination](#), page 98

Configuring T3 Controller Support for the Cisco AS5800

To configure T3 controller support specifically for the CT3 feature board in a Cisco AS5800 access server, use the following commands beginning in user EXEC mode.

SUMMARY STEPS

1. Router> **enable**
2. Router# **configure terminal**
3. Router(config)# **controller t3 shelf / slot / port**
4. Router(config-controller)# **description ascii-string**
5. Router(config-controller)# **cablelength number**
6. Router(config-controller)# **framing {c-bit | m23}**
7. Router(config-controller)# **t1 ds1 controller**
8. Router(config-controller)# **exit**
9. Router(config)# **controller t1 shelf / slot / port:t1-num**
10. Router(config-controller)# **exit**
11. Router(config)# **dial-tdm-clock priority number {external | trunk-slot number} ds3-port number port number**
12. Router(config)# **exit**
13. Router# **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>Router> enable</p> <p>Example:</p> <p>Password: <i>password</i></p> <p>Example:</p> <p>Router#</p>	Enters privileged EXEC mode.
Step 2	<p>Router# configure terminal</p> <p>Example:</p> <p>Enter configuration commands, one per line.</p> <p>Example:</p> <p>End with Ctrl-Z.</p> <p>Example:</p> <p>Router(config)#</p>	Enters global configuration mode.
Step 3	<p>Router(config)# controller t3 shelf / slot / port</p>	Enters controller configuration mode and specifies a shelf, slot, and port for the T3 controller. 0 is the only valid port value.
Step 4	<p>Router(config-controller)# description <i>ascii-string</i></p>	Allows the user to enter a description of the T3 controller.
Step 5	<p>Router(config-controller)# cablelength <i>number</i></p>	Specifies a controller cablelength value from 0 to 450(feet).
Step 6	<p>Router(config-controller)# framing { c-bit m23 }</p>	Specifies the type of T3 framing used: C-bit specifies c-bit parity framing; m23 (the default) specifies M23 multiplexer framing.
Step 7	<p>Router(config-controller)# t1 ds1 controller</p>	Creates a logical T1 controller from each of the specified T3 line time slots. <i>ds1</i> is a T1 time slot within the T3 line with a value from 1 to 28. (The T1 controller is in <i>shelf / slot /0:ds1</i> .)
Step 8	<p>Router(config-controller)# exit</p>	Exits controller configuration mode and returns to global configuration mode.

	Command or Action	Purpose
Step 9	Router(config)# controller t1 shelf / slot / port:t1-num	Enters controller configuration mode and specifies a port for the T1 controller. <i>t1-num</i> is a T1 time slot within the T3 line with a value from 1 to 28.
Step 10	Router(config-controller)# exit	Exits controller configuration mode and returns to global configuration mode.
Step 11	Router(config)# dial-tdm-clock priority number {external trunk-slot number} ds3-port number port number	Configures clock priority, which is a value from 1 to 50. Specifies a clocking source: either the external reference clock or any port of a trunk card. If you are using the external reference clock, no other CLI is needed. If you are using a trunk card, select a dial shelf slot from 0 to 5. Specifies a T3 port number, which has a value of 0. Possible T1 port values are from 1 to 28.
Step 12	Router(config)# exit	Returns to EXEC mode.
Step 13	Router# copy running-config startup-config	Saves your changes.

Configuring the T3 Controller

If you do not modify the configuration of the CT3IP, the configuration defaults shown in the table below are used.

Table 5 CT3IP Controller Defaults

Attribute	Default Value
Framing	auto-detect
Cable length	224 feet
Clock source	internal

If you must change any of the default configuration attributes, use the following commands beginning in global configuration mode.

Command	Purpose
Router(config)# controller t3 slot / port-adapter / port	Selects the CT3IP and enters controller configuration mode. The port adapter and port numbers for the CT3IP are 0.
Router(config-controller)# framing {c-bit m23 auto-detect}	(Optional) Changes the framing format.

Command	Purpose
Router(config-controller)# cablelength <i>feet</i>	(Optional) Changes the cable length (values are 0 to 450 feet). Although you can specify a cable length from 0 to 450 feet, the hardware only recognizes two ranges: 0 to 49 and 50 to 450. For example, entering 35 feet uses the 0 to 49 range. If you later change the cable length to 40 feet, there is no change because 40 is within the 0 to 49 range. However, if you change the cable length to 50, the 50 to 450 range is used. The actual number you enter is stored in the configuration file.
Router(config-controller)# clock source { internal line }	(Optional) Changes the clock source used by the T3 controller.

Configuring Each T1 Channel

You must configure the time slots used by each T1 channel on the CT3IP. Optionally, you can specify the speed, framing format, and clock source used by each T1 channel. If you do not specify the speed, framing format, and clock source used by each T1 channel, the configuration defaults shown in the table below are used.

Table 6 CT3IP T1 Channel Defaults

Attribute	Default Value
Speed	64 kbps
Framing	esf
Clock source	internal
Linecode	b8zs
T1 yellow alarm	detection and generation

To specify the time slots used by each T1 channel, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. Router(config)# **controller t3** *slot / port-adapter / port*
2. Router(config-controller)# **t1 channel timeslot range** [**speed** {**56** | **64**}]

DETAILED STEPS

Command or Action	Purpose
Step 1 Router(config)# controller t3 <i>slot / port-adapter / port</i>	Selects the CT3IP and enters controller configuration mode.
Step 2 Router(config-controller)# t1 channel timeslot <i>range [speed {56 64}]</i>	Configures the time slots (values are 1 to 24) for the T1 channel (values are 1 to 28) and optionally specifies the speed for each T1 channel.

**Note**

The 56-kbps speed is valid only for T1 channels 21 through 28.

**Note**

T1 channels on the CT3IP are numbered 1 to 28 rather than the more traditional zero-based scheme (0 to 27) used with other Cisco products. This numbering scheme is to ensure consistency with telco numbering schemes for T1 channels within channelized T3 equipment.

If you need to change any of the default configuration attributes, use the following commands, beginning in global configuration mode.

Command	Purpose
Router(config)# controller t3 <i>slot / port-adapter / port</i>	Selects the CT3IP and enters controller configuration mode.
Router(config-controller)# t1 channel framing { esf sf }	(Optional) Changes the framing format used by the T1 channel (values are 1 to 28). If you select sfframing , disable yellow alarm detection because the yellow alarm can be incorrectly detected with sfframing .
Router(config-controller)# no t1 channel yellow { detection generation }	(Optional) Disables detection or generation of a yellow alarm on the T1 channel (values are 1 to 28).
Router(config-controller)# t1 channel clock source { internal line }	(Optional) Changes the clock source used by the T1 channel (values are 1 to 28).
Router(config-controller)# t1 channel linecode { ami b8zs }	(Optional) Changes the line coding used by the T1 channel (values are 1 to 28). If you select ami line coding, you must also invert the data on the T1 channel by using the invert data interface command. To do so, first use the interface serial slot / port-adapter / port : t1-channel global configuration command to select the T1 channel and enter interface configuration mode.

After you configure the T1 channels on the CT3IP, you can continue configuring it as you would a normal serial interface. All serial interface commands might not be applicable to the T1 channel. For more information, see the [Configuring a Synchronous Serial Interface](#), page 66.

To enter interface configuration mode and configure the serial interface that corresponds to a T1 channel, use the following command in global configuration mode.

Command	Purpose
<code>Router(config)# interface serial slot / port-adapter / port:t1-channel</code>	Defines the serial interface for a T1 channel (values are 1 to 28) and enters interface configuration mode. The port adapter and port numbers for the CT3IP are 0.

In addition to the commands in the [Configuring a Synchronous Serial Interface](#), page 66, the **invert data** interface command can be used to configure the T1 channels on the CT3IP. If the T1 channel on the CT3IP is using AMI line coding, you must invert the data. For information on the **invert data** interface command, see the [Inverting the Data](#), page 80. For more information, refer to the **t1 linecode** controller configuration command in the *Cisco IOS Interface and Hardware Component Command Reference*.

Configuring External T1 Channels

The first three T1 channels (1, 2, and 3) of the CT3IP can be broken out to the DSUP-15 connectors so that the T1 channel can be further demultiplexed by the MIP on the same router, another router, or other multiplexing equipment.



Note

If a T1 channel that was previously configured as a serial interface is broken out to the external T1 port, that interface and its associated configuration remain intact while the channel is broken out to the external T1 port. The serial interface is not usable during the time that the T1 channel is broken out to the external T1 port; however, the configuration remains to facilitate the return of the T1 channel to a serial interface using the **no t1 external** command.

To configure a T1 channel as an external port, use the following commands beginning in EXEC mode.

SUMMARY STEPS

1. Router# **show controller t3** slot / port-adapter / port
2. Router# **configure terminal**
3. Router(config)# **controller t3** slot / port-adapter / port
4. Router(config-controller)# **t1 external** channel [cablelength feet] [linecode {ami | b8zs}]

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# show controller t3 slot / port-adapter / port	Displays the Ext1... field so that you can verify whether the external device connected to the external T1 port is configured and cabled correctly. If the line status is OK, a valid signal is being received and the signal is not an all-ones signal.
Step 2	Router# configure terminal	Enters global configuration mode.

Command or Action	Purpose
Step 3 Router(config)# controller t3 slot / port-adapter / port	Selects the CT3IP and enters controller configuration mode.
Step 4 Router(config-controller)# t1 external channel [cablelength feet] [linecode {ami b8zs}]	<p>Configures the T1 channel (values are 1, 2, and 3) as an external port and optionally specifies the cable length and line code. Only T1 channels 1 through 3 can be configured as an external T1.</p> <p>The default cablelength is 133 feet, and the default linecode is b8zs. Although you can specify a cable length from 0 to 655 feet, the hardware only recognizes the following ranges: 0 to 133, 134 to 266, 267 to 399, 400 to 533, and 534 to 655. For example, entering 150 feet uses the 134 to 266 range. If you later change the cable length to 200 feet, there is no change because 200 is within the 134 to 266 range. However, if you change the cable length to 399, the 267 to 399 range is used. The actual number you enter is stored in the configuration file.</p>

After you configure the external T1 channel, you can continue configuring it as a channelized T1 from the MIP. All channelized T1 commands might not be applicable to the T1 interface. To define the T1 controller and enter controller configuration mode, use the following command in global configuration mode.

Command	Purpose
Router(config)# controller t1 slot / port	Selects the MIP and enters controller configuration mode.

After you configure the channelized T1 on the MIP, you can continue configuring it as you would a normal serial interface. All serial interface commands might not be applicable to the T1 interface. To enter interface configuration mode and configure the serial interface that corresponds to a T1 channel group, use the following command in global configuration mode.

Command	Purpose
Router(config)# interface serial slot / port:t1-channel	Defines the serial interface for a T1 channel on the MIP (values are 1 to 28) and enters interface configuration mode.

For more information, see the [Configuring Each T1 Channel](#), page 89 and the [Specifying a Synchronous Serial Interface](#), page 67.

For an example of configuring an external T1 channel, see the [Channelized T3 Interface Processor Configuration Examples](#), page 143.

Monitoring and Maintaining the CT3IP

After configuring the new interface, you can monitor the status and maintain the CT3IP in the Cisco 7000 series routers with an RSP7000 or in the Cisco 7500 series routers by using the **show** commands. To display the status of any interface, use one of the following commands in EXEC mode.

Command	Purpose
Router> show controller cbus	Displays the internal status of each interface processor and lists each interface.

Command	Purpose
<pre>Router> show controller t3 [slot / port-adapter / port[: t1-channel]] [brief tabular]</pre>	<p>Displays the status of the T3 and T1 channels (values are 1 to 28), including the T3 alarms and T1 alarms for all 28 T1 channels, or only the T1 channel specified.</p>
<pre>Router> show interfaces serial slot / port-adapter / port : t1-channel [accounting crb]</pre>	<p>Displays statistics about the serial interface for the specified T1 channel (values are 1 to 28) on the router.</p>

Verifying T3 Configuration

To verify your software configuration, you can use **show** commands for controller settings. To use **show** commands, you must be in privileged EXEC mode.

```
Router#
show controller t3
T3 1/0/0 is up.
  Applique type is Channelized T3
  No alarms detected.
  FEAC code received: No code is being received
  Framing is M23, Line Code is B3ZS, Clock Source is Line.
  Data in current interval (751 seconds elapsed):
    0 Line Code Violations, 0 P-bit Coding Violation
    0 C-bit Coding Violation, 0 P-bit Err Secs
    0 P-bit Severely Err Secs, 0 Severely Err Framing Secs
    0 Unavailable Secs, 0 Line Errored Secs
    0 C-bit Errored Secs, 0 C-bit Severely Errored Secs
  Total Data (last 16 15 minute intervals):
    0 Line Code Violations, 0 P-bit Coding Violation,
    0 C-bit Coding Violation, 0 P-bit Err Secs,
    0 P-bit Severely Err Secs, 0 Severely Err Framing Secs,
    0 Unavailable Secs, 0 Line Errored Secs,
    0 C-bit Errored Secs, 0 C-bit Severely Errored Secs
```

**Tip**

To use the controller, it must report being up.

- No errors should be reported.

Configuring Maintenance Data Link Messages

The CT3IP can be configured to send a Maintenance Data Link (MDL) message as defined in the ANSI T1.107a-1990 specification. To specify the transmission of the MDL messages, use the following commands beginning in global configuration mode.

SUMMARY STEPS

- Router(config)# **controllers t3** *slot / port-adapter / port*
- Router(config-controller)# **mdl** {**transmit** {**path** | **idle-signal** | **test-signal**} | **string** {**eic** | **lic** | **fic** | **unit** | **pfi** | **port** | **generator**} *string*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config)# controllers t3 <i>slot / port-adapter / port</i>	Selects the CT3IP and enters controller configuration mode.
Step 2	Router(config-controller)# mdl { transmit { path idle-signal test-signal } string { eic lic fic unit pfi port generator } <i>string</i> }	Configures the MDL message.

Specify one **mdl** command for each message. For example, use **mdl string eic Router A** to transmit "Router A" as the equipment identification code and use **mdl string lic Test Network** to transmit "Test Network" as the location identification code.

Use the **show controllers t3** command to display MDL information (received strings). MDL information is displayed only when framing is set to C-bit.

Enabling Performance Report Monitoring

The CT3IP supports performance reports via the Facility Data Link (FDL) per ANSI T1.403. By default, performance reports are disabled. To enable FDL performance reports, use the following commands beginning in global configuration mode.

SUMMARY STEPS

- Router(config)# **controllers t3** *slot / port-adapter / port*
- Router(config-controller)# **t1 channel fdl ansi**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config)# controllers t3 <i>slot / port-adapter / port</i>	Selects the CT3IP and enters controller configuration mode.

Command or Action	Purpose
Step 2 Router(config-controller)# t1 channel fdl ansi	Enables 1-second transmission of the performance report for a specific T1 channel (values are 1 to 28).



Note

Performance reporting is available only on T1 channels configured for ESF framing.

To display the remote performance report information, use the following command in EXEC mode.

Command	Purpose
Router> show controllers t3 [<i>slot / port-adapter / port[: t1-channel]</i>] remote performance [brief tabular]	Displays the remote performance report information for the T1 channel (values are 1 to 28).

Configuring for BERT on the Cisco AS5300

Bit-error rate testing (BERT) and loopbacks are used by carriers and Internet service providers (ISPs) to aid in problem resolution as well as testing the quality of T1/E1 links. BERT detects poor quality links early and isolates problems quickly, enabling Cisco AS5300 users to improve their quality of service and increase their revenues.

BERT is available for the Cisco AS5300 router for T1 and E1 facilities. Perform the following tasks to configure the Cisco AS5300 router for BERT, use the following commands beginning in user EXEC mode.

SUMMARY STEPS

1. 5300> **enable**
2. 5300# **configure terminal**
3. Router(config)# **bert profile**

DETAILED STEPS

Command or Action	Purpose
Step 1 5300> enable Example: Password: <i>password</i> Example: 5300#	Enters privileged EXEC mode.
Step 2 5300# configure terminal	Enters global configuration mode.

Command or Action	Purpose
Step 3 Router(config)# bert profile	Enables the user to configure up to 15 BERT profiles in addition to the default BERT profile 0, by using the extensions to this command.

Verifying BERT on the Cisco AS5300

To verify that a BERT feature is running, use the **show running-config** command in EXEC mode.

```
5300> show running-config
!
bert profile 1 pattern 1s threshold 10^-4 error-injection none duration 3
bert profile 7 pattern 220-0.151QRSS threshold 10^-3 error-injection 10^-5 duration 120
```

Enabling a BERT Test Pattern

To enable and disable generation of a BERT test pattern for a specified interval for a specific T1 channel, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. Router(config)# **controller t3 slot / port-adapter / port**
2. Router(config-controller)# **t1 channel bert pattern {0s | 1s | 2^15 | 2^20 | 2^23} interval minutes**
3. Router(config-controller)# **no t1 channel bert pattern {0s | 1s | 2^15 | 2^20 | 2^23} interval minutes**

DETAILED STEPS

Command or Action	Purpose
Step 1 Router(config)# controller t3 slot / port-adapter / port	Selects the CT3IP and enters controller configuration mode.
Step 2 Router(config-controller)# t1 channel bert pattern {0s 1s 2^15 2^20 2^23} interval minutes	Enables a BERT test pattern on a T1 channel (values are 1 to 28).
Step 3 Router(config-controller)# no t1 channel bert pattern {0s 1s 2^15 2^20 2^23} interval minutes	Disables a BERT test pattern on a T1 channel (values are 1 to 28).

The BERT test patterns from the CT3IP are framed test patterns (that is, the test patterns are inserted into the payload of the framed T1 signal).

To view the BERT results, use the **show controllers t3** or **show controllers t3 brief** EXEC command. The BERT results include the following information:

- Type of test pattern selected
- Status of the test
- Interval selected
- Time remaining on the BERT test
- Total bit errors
- Total bits received

When the T1 channel has a BERT test running, the line state is DOWN. Also, when the BERT test is running and the Status field is Not Sync, the information in the total bit errors field is not valid. When the BERT test is done, the Status field is not relevant.

The **t1 bert pattern** command is not written to NVRAM because it is only used for testing the T1 channel for a short predefined interval and to avoid accidentally saving the command, which could cause the interface not to come up the next time the router reboots.

Enabling Remote FDL Loopbacks

You can perform the following types of remote Facility Data Link (FDL) loopbacks on a T1 channel:

- Remote payload FDL ANSI--Sends a repeating, 16-bit Extended Superframe (ESF) data link code word (00010100 11111111) to the remote end requesting that it enter into a network payload loopback.
- Remote line FDL ANSI--Sends a repeating, 16-bit ESF data link code word (00001110 11111111) to the remote CSU end requesting that it enter into a network line loopback.
- Remote line FDL Bellcore--Sends a repeating, 16-bit ESF data link code word (00010010 11111111) to the remote SmartJack end requesting that it enter into a network line loopback.

To enable loopback on a T1 channel, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. Do one of the following:
 - Router(config)# **interface serial** *slot / port-adapter / port:t1-channel*
 - (Cisco 7500 series and Cisco 7000 series routers with the RSP7000 and RSP7000CI)
 -
 - Router(config)# **interface serial** *slot / port:t1-channel*
2. Router(config-if)# **loopback remote payload** [**fdl**] [**ansi**]
3. Router(config-if)# **loopback remote line fdl** {**ansi** | **bellcore**}

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 Do one of the following:</p> <ul style="list-style-type: none"> • Router(config)# interface serial <i>slot / port-adapter / port:t1-channel</i> • (Cisco 7500 series and Cisco 7000 series routers with the RSP7000 and RSP7000CI) • • Router(config)# interface serial <i>slot / port:t1-channel</i> <p>Example:</p> <pre>(Cisco 7200 series)</pre>	<p>Selects the T1 channel (values are 1 to 28) on the CT3IP and enters interface configuration mode.</p>

Command or Action	Purpose
Step 2 Router(config-if)# loopback remote payload [fdl] [ansi]	Enables the remote payload FDL ANSI bit loopback on the T1 channel.
Step 3 Router(config-if)# loopback remote line fdl {ansi bellcore}	Enables the remote line FDL ANSI bit loopback or remote SmartJack loopback on the T1 channel.



Note The port adapter and port numbers for the CT3IP are 0.

Configuring T1 Cable Length and T1 E1 Line Termination

When you configure your channelized T1 trunk cards, you can change the line build-out of the cable pair connected to the port. To specify the build-out value, use either the **cablelength long** command or the **cablelength short** command. These commands are not required for E1 trunk cards.

For cables longer than 655 feet, use the **cablelength long** command; for cables up to and including 655 feet, use the **cablelength short** command.

The **line-termination** command allows you to set the T1/E1 port termination to 75 ohms unbalanced or 120 ohms balanced.

The following cable length short configurations define the length range (in feet) between your network access server (NAS) and your repeater. The **cablelength short** command is configured for a channelized T1 only and includes the following settings:

- 133 feet (0 to 133 feet)
- 266 feet (134 to 266 feet)
- 399 feet (267 to 399 feet)
- 533 feet (400 to 533 feet)
- 655 feet (534 to 655 feet)



Note Although you can specify a cable length from 0 to 655 feet, the hardware only recognizes fixed configuration lengths. For example, if your cable length is 50 feet between your NAS and your repeater, you should configure your cable length using the 133-foot setting. If you later change the cable length to 200 feet, you should reconfigure your cable length using the 266-foot setting.

The following cable length long configurations define the length range in gain and pulse requirements for the length of build-out between your NAS and your repeater that is longer than 655 feet. The **cablelength long** command is configured for a channelized T1 only and includes the following gain and pulse settings:

- gain26 (26 dB gain)
- gain36 (36 dB gain)
- -15db (-15 dB pulse)
- -22.5db (-22.5 dB pulse)
- -7.5db (-7.5 dB pulse)
- 0db (0 dB pulse)

To configure channelized T1 lines for line build-out, use the following commands beginning in user EXEC mode.

SUMMARY STEPS

1. Router> **enable**
2. Router# **configure terminal**
3. Router(config)# **controller t1 shelf / slot / port**
4. Do one of the following:
 - Router(config-controller)# **cablelength short(133| 266| 399| 533| 655)**
 -
 -
 - Router(config-controller)# **cablelength long {gain26 | gain36} {-15 | -22.5 | -7.5 | 0}**
5. Router(config-controller)# **line termination {75-ohm | 120-ohm}**

DETAILED STEPS

Command or Action	Purpose
Step 1 Router> enable Example: Password: <i>password</i> Example: Router#	Enters privileged EXEC mode.
Step 2 Router# configure terminal	Enters global configuration mode. The example shown uses the terminal configuration option.
Step 3 Router(config)# controller t1 shelf / slot / port	Enters controller configuration mode and specifies a shelf, slot, and port for the controller port. The controller ports are labeled <i>shelf / slot / 0</i> through <i>shelf / slot / 11</i> on the T1. (You must type in the slashes (/) as part of the command.)
Step 4 Do one of the following: <ul style="list-style-type: none"> • Router(config-controller)# cablelength short(133 266 399 533 655) • • • Router(config-controller)# cablelength long {gain26 gain36} {-15 -22.5 -7.5 0} 	Specifies the controller cablelength short value between 0 and 655 (feet). Specifies the controller cablelength long value using gain and pulse settings for cables longer than 655 feet. (Configure cable length for T1 only.)
Step 5 Router(config-controller)# line termination {75-ohm 120-ohm}	Specifies the line-termination value. (The command is used for E1 only.)

Configuring PA-E3 and PA-2E3 Serial Port Adapters

The PA-E3 and PA-2E3 serial port adapters are available on:

- Cisco 7200 series routers
- Cisco 7500 series routers
- Cisco 7000 series routers with the 7000 series Route Switch Processor (RSP7000) and 7000 series Chassis Interface (RSP7000CI)

These port adapters provide one (PA-E3) or two (PA-2E3) high-speed, full-duplex, synchronous serial E3 interfaces and integrated data service unit (DSU) functionality.

The E3 port adapters can transmit and receive data at E3 rates of up to 34 Mbps and use a 75-ohm coaxial cable available from Cisco to connect to a serial E3 network. These port adapters support the following:

- 16- and 32-bit cyclic redundancy checks (CRCs)
- High-speed HDLC data
- G.751 framing or bypass
- HDB3 line coding
- ATM-DXI, Frame Relay, HDLC, PPP, and SMDS serial encapsulation
- National service bits
- E3 MIB (RFC 1407)
- Scrambling and reduced bandwidth
- Remote and local loopbacks

The PA-E3 port adapter supports the RFC 1407 DS3 Near End Group, including:

- DS3/E3 Configuration Table
- DS3/E3 Current Table
- DS3/E3 Interval Table
- DS3/E3 Total Table

The PA-E3 port adapter also supports the Card Table in the Cisco Chassis MIB and the MIB-2 for each PA-E3 interface.

The PA-E3 port adapter does not support the RFC 1407 DS3 Far End Group and DS3/E3 Fractional Group.



Note

For additional information on the E3 serial port adapter, refer to the *PA-E3 Serial Port Adapter Installation and Configuration* publication.

- [PA-E3 and PA-2E3 Serial Port Adapter Configuration Task List, page 100](#)

PA-E3 and PA-2E3 Serial Port Adapter Configuration Task List

To configure the PA-E3, Perform the tasks in the following sections. Each task in the list is identified as either required or optional.

For PA-E3 port adapter configuration examples, see the [PA-E3 Serial Port Adapter Configuration Example, page 146](#).

- [Configuring the PA-E3 Port Adapter, page 101](#)

- [Monitoring and Maintaining the PA-E3 Port Adapter, page 102](#)

Configuring the PA-E3 Port Adapter

The commands listed in the table below have been added to support the PA-E3 interface configuration. If you do not modify the configuration of the PA-E3, the configuration defaults shown in the table below are used.

Table 7 PA-E3 Port Adapter Defaults

Command	Default Value
dsu bandwidth	34,010 kbps
dsu mode	0
framing	g751
international bit	0 0
invert data	data is not inverted
national bit	0
scramble	disabled

If you need to change any of the default configuration attributes, use the first command in global configuration mode, followed by any of the optional commands in interface configuration mode.

Command	Purpose
Router(config)# interface serial <i>slot / port-adapter / port</i>	Selects the PA-E3 interface and enters interface configuration mode for the Cisco 7500 series and Cisco 7000 series routers with the RSP7000 and RSP7000CI.
or	
Router(config)# interface serial <i>slot / port</i>	Selects the PA-E3 interface and enters interface configuration mode for the Cisco 7200 series.

Command	Purpose
Router(config-if)# dsu bandwidth <i>kbps</i>	Changes the DSU bandwidth.
Router(config-if)# dsu mode {0 1}	Changes the DSU mode. To connect to another PA-E3 port adapter or a Digital Link DSU, use the default mode (0). To connect to a Kentrox DSU, use mode 1.
Router(config-if)# framing {g751 bypass}	Changes the framing used by the interface.
Router(config-if)# international bit {0 1} {0 1}	Changes the international bit used by the interface.
Router(config-if)# invert data	Inverts the data stream on the interface.
Router(config-if)# national bit {0 1}	Changes the national bit used by the interface.
Router(config-if)# scramble	Enables scrambling on the interface.

Monitoring and Maintaining the PA-E3 Port Adapter

After configuring the new interface, you can display its status. To show current status of the E3 interface on the PA-E3 port adapter, use any of the following commands in EXEC mode.

Command	Purpose
Router> show interfaces serial <i>slot / port-adapter / port</i>	Displays statistics for the E3 interface for the Cisco 7500 series and Cisco 7000 series routers with the RSP7000 and RSP7000CI.
or	Displays statistics for the E3 interface for the Cisco 7200 series.
Router> show interfaces serial <i>slot / port</i>	
Router> show controllers serial <i>slot / port-adapter / port</i>	Displays the configuration information for the E3 interface for the Cisco 7500 series and Cisco 7000 series routers with the RSP7000 and RSP7000CI.
or	Displays the configuration information for the E3 interface for the Cisco 7200 series.
Router> show controllers serial <i>slot / port</i>	

Configuring PA-T3 and PA-2T3 Serial Port Adapters

The PA-T3 and PA-2T3 serial port adapters are available on:

- Cisco 7200 series routers

- Second-generation Versatile Interface Processor (VIP2) in all Cisco 7500 series routers
- Cisco 7000 series routers with the 7000 series Route Switch Processor (RSP7000) and 7000 series Chassis Interface (RSP7000CI)

These port adapters provide one (PA-T3) or two (PA-2T3) high-speed, full-duplex, synchronous serial T3 interfaces and integrated data service unit (DSU) functionality.

The T3 port adapters can transmit and receive data at T3 rates of up to 45 Mbps and use a 75-ohm coaxial cable available from Cisco to connect to a serial T3 network. These port adapters support the following features:

- 16- and 32-bit cyclic redundancy checks (CRCs)
- High-speed HDLC data
- C-bit, M13, and bypass framing
- HDB3 line coding
- ATM-DXI, Frame Relay, HDLC, PPP, and SMDS serial encapsulation
- DS3 MIB (RFC 1407)
- Scrambling and reduced bandwidth
- Remote and local loopbacks

**Note**

For additional information on interoperability guidelines for T3 serial port adapter DSUs, refer to the *PA-T3 Serial Port Adapter Installation and Configuration* publication.

- [PA-T3 and PA-2T3 Port Adapter Configuration Task List, page 103](#)

PA-T3 and PA-2T3 Port Adapter Configuration Task List

To configure the PA-T3 port adapters, perform the tasks in the following sections. Each task is identified as either required or optional.

For PA-T3 port adapter configuration examples, see the [PA-T3 and PA-2T3 Configuration Example, page 147](#).

- [Configuring the PA-T3 Port Adapter, page 103](#)
- [Troubleshooting the PA-T3 Port Adapter, page 105](#)
- [Monitoring and Maintaining the PA-T3 Port Adapter, page 105](#)

Configuring the PA-T3 Port Adapter

The commands listed in the table below have been added to support the PA-T3 interface configuration. If you do not modify the configuration of the PA-T3, the configuration defaults shown in the table below are used.

Table 8 PA-T3 Port Adapter Defaults

Command	Default Value
cablelength	49
clock source	line
crc 32	16-bit
dsu bandwidth	44,736 kbps
dsu mode	0
framing	C-bit
invert data	data is not inverted
scramble	disabled

If you need to change any of the default configuration attributes, use the first command in global configuration mode, followed by any of the optional commands in interface configuration mode.

Command	Purpose
Router(config)# interface serial <i>slot / port-adapter / port</i>	Selects the PA-T3 interface and enters interface configuration mode for the Cisco 7500 series and Cisco 7000 series routers with the RSP7000 and RSP7000CI.
OR	Selects the PA-T3 interface and enters interface configuration mode for the Cisco 7200 series.
Router(config)# interface serial <i>slot / port</i>	
Router(config-if)# cablelength <i>length</i>	Changes the cable length.

Command	Purpose
Router(config-if)# crc 32	Enables 32-bit CRC.
Router(config-if)# dsu bandwidth <i>kbps</i>	Changes the DSU bandwidth.
Router(config-if)# dsu mode {0 1 2}	Changes the DSU mode. To connect to another PA-T3 port adapter or a Digital Link DSU, use the default mode (0). To connect to a Kentrox DSU, use mode 1. To connect to a Larscom DSU, use mode 2.
Router(config-if)# framing {c-bit m13 bypass}	Changes the framing used by the interface.
Router(config-if)# invert data	Inverts the data stream on the interface.
Router(config-if)# scramble	Enables scrambling on the interface.

Troubleshooting the PA-T3 Port Adapter

To set the following loopback modes to troubleshoot the PA-T3 port adapter using Cisco IOS software, use the first command in global configuration mode, followed by any of the other commands depending on your needs:

Command	Purpose
Router(config)# loopback dte	Loops back after the LIU toward the terminal.
Router(config)# loopback local	Loops back after going through the framer toward the terminal.
Router(config)# loopback network line	Loops back toward the network before going through the framer.
Router(config)# loopback network payload	Loops back toward the network after going through the framer.
Router(config)# loopback remote	Sends a far-end alarm control (FEAC) to set the remote framer in loopback.

These loopback commands loop all packets from the T3 interface back to the interface or direct packets from the network back out toward the network.

Monitoring and Maintaining the PA-T3 Port Adapter

After configuring the new interface, you can display its status. To show current status of the T3 interface on the PA-T3 port adapter, use any of the following commands in EXEC mode.

Command	Purpose
Router> show version	Displays system hardware configuration.
Router> show controllers cbus	Displays current interface processors and their interfaces.
Router> show interfaces <i>slot / port-adapter / port</i> (Cisco 7500 series and Cisco 7000 series routers with the RSP7000 and RSP7000CI)	Displays statistics for the T3 interface.
Router> show interfaces <i>slot / port</i> (Cisco 7200 series)	
Router> show controllers serial <i>slot / port-adapter / port</i> (Cisco 7500 series and Cisco 7000 series routers with the RSP7000 and RSP7000CI)	Displays the configuration information for the T3 interface.
Router> show controllers serial <i>slot / port</i> (Cisco 7200 series)	
Router> show protocols	Displays protocols configured for the system and specific interfaces.
Router> more system:running-config	Displays the running configuration file.
Router> more nvram:startup-config	Displays the configuration stored in NVRAM.
Router> show diag <i>slot</i>	Displays specific port adapter information

Configuring a Packet OC-3 Interface

The Cisco Packet OC-3 Interface Processor (POSIP) and Packet OC-3 Port Adapter (POSPA) are available on:

- Cisco 7500 series routers
- Cisco 7200 series routers

The Packet-Over-SONET OC-3 port adapters (PA-POS-OC3SML, PA-POS-OC3SMI, and PA-POS-OC3MM) are available on:

- Cisco 7500 series routers
- Cisco 7200 series routers
- Cisco 7000 series routers with the 7000 Series Route Switch Processor (RSP7000) and 7000 Series Chassis Interface (RSP7000CI)

The POSIP and POS OC-3 provide a single 155.520-Mbps, OC-3 physical layer interface for packet-based traffic. This OC-3 interface is fully compatible with SONET and Synchronous Digital Hierarchy (SDH) network facilities and is compliant with RFC 1619, "PPP over SONET/SDH," and RFC 1662, *PPP in HDLC-like Framing*. The Packet-Over-SONET specification is primarily concerned with the use of PPP encapsulation over SONET/SDH links.

For more information on the PA-POS-OC3 port adapter, refer to the *PA-POS-OC3 Packet OC-3 Port Adapter Installation and Configuration* publication that accompanies the hardware.

The POS is a fixed-configuration interface processor that uses second-generation Versatile Interface Processor (VIP2) technology. The POS provides a single 155.520-Mbps, OC-3 physical layer interface for packet-based traffic. This OC-3 interface is fully compatible with SONET and SDH network facilities and is compliant with RFC 1619 and RFC 1662. The Packet-Over-SONET specification primarily addresses the use of PPP encapsulation over SONET/SDH links.

The table below describes the default values set in the initial configuration of a Packet OC-3 interface.

Table 9 Packet OC-3 Interface Default Configuration

Attributes	Default Value
Maximum transmission unit (MTU)	4470 bytes
Framing	SONET STS-3c framing
Loopback internal	No internal loopback
Loopback line	No line loopback
Transmit clocking	Recovered receive clock
Enabling	Shut down

Because the Packet OC-3 interface is partially configured, you might not need to change its configuration before enabling it. However, when the router is powered up, a new Packet OC-3 interface is shut down. To enable the Packet OC-3 interface, you must use the **no shutdown** command in the global configuration mode.

- [Packet OC-3 Interface Configuration Task List, page 107](#)

Packet OC-3 Interface Configuration Task List

The values of all Packet OC-3 configuration parameters can be changed to match your network environment. To customize the POS configuration, perform the tasks in the following sections. Each task in the list is identified as either required or optional.

- [Selecting a Packet OC-3 Interface, page 108](#)
- [Setting the MTU Size, page 108](#)
- [Configuring Framing, page 108](#)
- [Configuring an Interface for Internal Loopback, page 109](#)
- [Configuring an Interface for Line Loopback, page 109](#)
- [Setting the Source of the Transmit Clock, page 109](#)
- [Enabling Payload Scrambling, page 109](#)
- [Configuring an Alarm Indication Signal, page 110](#)

Selecting a Packet OC-3 Interface

The Packet OC-3 interface is referred to as *pos* in the configuration commands. An interface is created for each POS found in the system at reset time.

If you need to change any of the default configuration attributes or otherwise reconfigure the Packet OC-3 interface, use one the following commands in global configuration mode.

Command	Purpose
Router(config)# interface pos <i>slot / port</i> (Cisco 7200)	Selects the Packet OC-3 interface and enters interface configuration mode.
or	
Router(config)# interface pos <i>slot / port-adapter / port</i> (Cisco 7500)	

Setting the MTU Size

To set the maximum transmission unit (MTU) size for the interface, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# mtu <i>bytes</i>	Sets the MTU size.

The value of the *bytes* argument is in the range 64 to 4470 bytes; the default is 4470 bytes (4470 bytes exactly matches FDDI and HSSI interfaces for autonomous switching). The **no** form of the command restores the default.



Caution

Changing an MTU size on a Cisco 7500 series router will result in resizing and reassignment of buffers and resetting of all interfaces. The following message is displayed: %RSP-3-Restart:cbus complex88

Configuring Framing

To configure framing on the Packet OC-3 interface, use one of the following commands in interface configuration mode.

Command	Purpose
Router(config-if)# pos framing-sdh	Selects SDH STM-1 framing.
Router(config-if)# no pos framing-sdh	Reverts to the default SONET STS-3c framing.

Configuring an Interface for Internal Loopback

With the **loopback internal** command, packets from the router are looped back in the framer. Outgoing data gets looped back to the receiver without actually being transmitted. With the **loopback line** command, the receive fiber (RX) is logically connected to the transmit fiber (TX) so that packets from the remote router are looped back to it. Incoming data gets looped around and retransmitted without actually being received.

To enable or disable internal loopback on the interface, use one of the following commands in interface configuration mode.

Command	Purpose
<code>Router(config-if)# loop internal</code>	Enables internal loopback.
<code>Router(config-if)# no loop internal</code>	Disables internal loopback.

Local loopback is useful for checking that the POS is working. Packets from the router are looped back in the framer.

Configuring an Interface for Line Loopback

Line loopback is used primarily for debugging purposes.

To enable or disable an interface for line loopback, use one of the following commands in interface configuration mode.

Command	Purpose
<code>Router(config-if)# loop line</code>	Enables line loopback.
<code>Router(config-if)# no loop line</code>	Disables line loopback.

The receive fiber (RX) is logically connected to the transmit fiber (TX) so that packets from the remote router are looped back to it.

Setting the Source of the Transmit Clock

By default, the Packet OC-3 interface uses the recovered receive clock to provide transmit clocking. To change the transmit clock source, use one of the following commands in interface configuration mode.

Command	Purpose
<code>Router(config-if)# clock source</code>	Sets the internal clock as the transmit clock source.
<code>Router(config-if)# no clock source</code>	Sets the recovered receive clock to provide transmit clocking.

Enabling Payload Scrambling

SONET payload scrambling applies a self-synchronous scrambler (x43+1) to the Synchronous Payload Envelope (SPE) of the interface to ensure sufficient bit transition density. Both ends of the connection must use the same scrambling algorithm. When enabling POS scrambling on a VIP2 POS on the Cisco 7500 series router that has a hardware revision of 1.5 or higher, you can specify CRC 16 only (that is, CRC 32 is currently not supported).

To enable SONET payload scrambling on a POS interface, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# pos scramble-atm</code>	Enables SONET payload scrambling.

Configuring an Alarm Indication Signal

To configure line alarm indication signals (LAIS) when the POS interface is placed in any administrative shut down state, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# pos ais-shut</code>	Sends line alarm indication signals.

Configuring a DPT OC-12c Interface

The dual-width OC-12c Dynamic Packet Transport (DPT) port adapter is available on Cisco 7200 series routers and Cisco 7200 VXR series routers with the correct Route Switch Processor (RSP2 or RSP4), running Cisco IOS Release 12.0(6)S or later, to provide shared IP-over-SONET capability.

The OC-12c Dynamic Packet Transport Interface Processor (DPTIP) is available on Cisco 7500 series routers with the correct Route Switch Processor (RSP2 or RSP4), running Cisco IOS Release 12.0(6)S or later. The DPT is an OC-12c interface that uses second-generation Versatile Interface Processor (VIP2) technology to provide shared IP-over-SONET capability, and it complies with IEEE 802.3 specifications for multicast and broadcast media. The DPTIP assembly consists of a VIP2 with a dual-width DPT interface processor permanently attached to it.

The DPT interface provides the following benefits:

- Accommodates large-scale network topology.
- Complies with applicable IEEE 802.3 standards.
- Supports Intelligent Protection Switching (IPS).

The interface type of the DPT or DPTIP is Spatial Reuse Protocol (SRP). SRP is a Cisco-developed MAC-layer protocol, used in conjunction with Cisco's DPT product family. DPT products deliver scalable Internet service, reliable IP-aware optical transport, and simplified network operations. These solutions allow you to scale and distribute your IP services across a reliable optical packet ring infrastructure.

Spatial bandwidth reuse is possible due to the packet destination-stripping property of SRP. Older technologies incorporate source stripping, where packets traverse the entire ring until they are removed by the source. Even if the source and destination nodes are next to each other on the ring, packets continue to traverse the entire ring until they return to the source to be removed. SRP provides more efficient use of available bandwidth by having the destination node remove the packet after it is read. This provides more bandwidth for other nodes on the SRP ring.

SRP rings consists of two counterrotating fibers, known as outer and inner rings, both concurrently used to carry data and control packets. SRP uses both explicit control packets and control information piggybacked inside data packets (control packets handle tasks such as keepalives, protection switching, and bandwidth control propagation). Control packets propagate in the opposite direction from the corresponding data packets, ensuring that the data takes the shortest path to its destination. The use of dual fiber-optic rings provides a high level of packet survivability. In the event of a failed node or a fiber cut, data is transmitted over the alternate ring.

SRP rings are media independent and can operate over a variety of underlying technologies, including SONET/SDH, wavelength division multiplexing (WDM), and dark fiber. This ability to run SRP rings over any embedded fiber transport infrastructure provides a path to packet-optimized transport for high-bandwidth IP networks.

One of the important benefits of SRP is the bandwidth scalability and efficiency with growth opportunities from OC-12c/STM-4c rings up to OC-192c/STM-64c rings.

- [OC-12c Interface Configuration Task List, page 111](#)

OC-12c Interface Configuration Task List

To configure the DPT interface, perform the tasks in the following sections. Each task in the list is identified as either required or optional.

- [Configuring the Dynamic Packet Transport Interface, page 111](#)
- [Configuring Intelligent Protection Switching, page 112](#)
- [Configuring DPT Topology, page 113](#)

Configuring the Dynamic Packet Transport Interface

SUMMARY STEPS

1. Router# **show running-config**
2. Router# **configure terminal**
3. Router(config)# **ip routing**
4. Router(config)# **interface srp slot / port**
5. Router(config-if)# **ip address ip-address mask**
6. Add any additional configuration commands required to enable routing protocols, and set the interface characteristics for your configuration requirements.
7. Router(config-if)# **no shutdown**
8. Router(config-if)# **exit**
9. Router# **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# show running-config	Confirms that the system recognizes the DPT or DTPIP.
Step 2	Router# configure terminal	Enters configuration mode.

	Command or Action	Purpose
Step 3	Router(config)# ip routing	Enables IP routing.
Step 4	Router(config)# interface srp slot / port Example: (Cisco 7200 series router) Example: Router(config)# interface srp slot / port-adapter / port Example: (Cisco 7500 series router)	Specifies an interface. The interface type of the DPT or DPTIP is Spatial Reuse Protocol (SRP).
Step 5	Router(config-if)# ip address ip-address mask	Assigns an IP address and subnet mask to the interface.
Step 6	Add any additional configuration commands required to enable routing protocols, and set the interface characteristics for your configuration requirements.	
Step 7	Router(config-if)# no shutdown	Changes the shutdown state to up and enables the interface.
Step 8	Router(config-if)# exit	Includes all the configuration commands to complete the configuration and exits interface configuration mode.
Step 9	Router# copy running-config startup-config	Writes the new configuration to the startup configuration.

The system displays an OK message when the configuration has been stored.

Use the **show running-config** command to verify the currently running configuration. Use the **show version** command to display the configuration of the system hardware and the Cisco IOS software information.

Configuring Intelligent Protection Switching

The SRP interface uses ring architecture to provide redundancy and protection from a failed node or a fiber cut by using Intelligent Protection Switching (IPS). To configure IPS, use the following commands beginning in privileged EXEC mode. The steps described in this section are optional.

SUMMARY STEPS

1. Router# **configure terminal**
2. Router(config)# **interface srp slot / port**
3. Router(config-if)# **srp ips request manual-switch a**
4. Router(config-if)# **srp ips wtr-timer 10**
5. Router(config-if)# **srp ips timer 20 a**
6. Router(config-if)# **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# configure terminal	Enables configuration mode.
Step 2	Router(config)# interface srp slot / port Example: (Cisco 7200 series routers) Example: Router(config)# interface srp slot / port-adapter / port Example: (Cisco 7500 series routers)	Configures an SRP interface.
Step 3	Router(config-if)# srp ips request manual-switch a	Specifies an IPS manual switch on side A or side B.
Step 4	Router(config-if)# srp ips wtr-timer 10	Specifies a wait-to-restore request (in seconds) to prevent switch oscillations on side A.
Step 5	Router(config-if)# srp ips timer 20 a	Configures a message timer to be sent to the inner and outer rings to control the frequency of IPS message transmissions on side A.
Step 6	Router(config-if)# exit	Exits configuration mode.

Use the **show srp** command to verify the configuration.

Configuring DPT Topology

Every node on a DPT ring maintains a topology map of the ring, so that it knows where to route traffic. It updates the topology map by periodically sending a query, called a topology discovery packet, out onto the outer-ring path. Each node on the ring adds its own MAC address to the packet. When the discovery packet returns to the originating node, the contents of the packet are used to update the topology map. You use the **srp topology-timer** command to set the frequency with which the node sends out topology discovery packets. To configure DPT, use the following commands beginning in privileged EXEC mode.

SUMMARY STEPS

1. Router# **configure terminal**
2. Router(config)# **interface srp slot / port**
3. Router(config-if)# **srp topology-timer 70**
4. Router(config-if)# **exit**
5. Router# **show srp topology**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# configure terminal	Enables interface configuration mode.
Step 2	Router(config)# interface srp slot / port Example: (Cisco 7200 series routers) Example: Router(config)# interface srp slot / port-adapter / port Example: (Cisco 7500 series routers)	Configures an SRP interface.
Step 3	Router(config-if)# srp topology-timer 70	Configures the frequency of the topology message timer in seconds.
Step 4	Router(config-if)# exit	Exits configuration mode.
Step 5	Router# show srp topology	Confirms the identity of the nodes on the ring; shows the number of hops between nodes; identifies the nodes that are in wrap mode. Use the show srp topology command to show the identity of the nodes on the DPT ring according to their MAC addresses.

Configuring Automatic Protection Switching of Packet-over-SONET Circuits

The automatic protection switching (APS) feature is supported on Cisco 7500 series routers. This feature allows switchover of Packet-over-SONET (POS) circuits and is often required when connecting SONET equipment to telco equipment. APS refers to the mechanism of bringing a "protect" POS interface into the SONET network as the "working" POS interface on a circuit from the intervening SONET equipment.

The protection mechanism used for this feature is "1+1, Bidirectional, nonrevertive" as described in the Bellcore publication "TR-TSY-000253, SONET Transport Systems; Common Generic Criteria, Section

5.3." In the 1+1 architecture, there is one working interface (circuit) and one protect interface, and the same payload from the transmitting end is sent to both the receiving ends. The receiving end decides which interface to use. The line overhead (LOH) bytes (K1 and K2) in the SONET frame indicate both status and action.

The protect interface is configured with the IP address of the router that has the working interface. The APS Protect Group Protocol, which runs on top of UDP, provides communication between the process controlling the working interface and the process controlling the protect interface. Using this protocol, POS interfaces can be switched because of a router failure, degradation or loss of channel signal, or manual intervention. In bidirectional mode, the receive and transmit channels are switched as a pair. In unidirectional mode, the transmit and receive channels are switched independently. For example, if the receive channel on the working interface has a loss of channel signal, both the receive and transmit channels are switched.

In addition to the new Cisco IOS commands added for the APS feature, the POS interface configuration commands **pos threshold** and **pos report** have been added to support user configuration of the bit-error rate (BER) thresholds and reporting of SONET alarms.

- [APS Configuration Task List, page 115](#)

APS Configuration Task List

Two SONET connections are required to support APS. In a telco environment, the SONET circuits must be provisioned as APS. You must also provision the operation (for example, 1+1), mode (for example, bidirectional), and revert options (for example, no revert). If the SONET connections are homed on two separate routers (the normal configuration), an out of band (OOB) communications channel between the two routers needs to be set up for APS communication.

When configuring APS, we recommend that you configure the working interface first. Normal operation with 1+1 operation is to configure it as a working interface. Also configure the IP address of the interface being used as the APS OOB communications path.

For more information on POS interfaces, refer to the installation and configuration documentation that accompanies the POS hardware.

To configure APS and POS, perform the following tasks. Each task is identified as either required or optional.

- [Configuring APS Working and Protect Interfaces, page 115](#)
- [Configuring APS Working and Protect Interfaces, page 116](#)
- [Configuring Other APS Options, page 117](#)
- [Monitoring and Maintaining APS, page 118](#)
- [Configuring SONET Alarm Reporting, page 118](#)
- [Configuring a Protection Switch, page 118](#)

Configuring APS Working and Protect Interfaces

This section describes how to configure and protect a working interface. The commands listed in this section are required. To avoid having the protected interface become the active circuit and disabling the working circuit when it is discovered, configure the working interface before configuring the protected interface.

To configure the working interface, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. Router(config)# **interface pos slot / port-adapter / port**
2. Router(config-if)# **aps working circuit-number**
3. Router(config-if)# **end**
4. Router# **show controllers pos**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config)# interface pos slot / port-adapter / port	Specifies the POS interface to be configured as the working interface and enters interface configuration mode.
Step 2	Router(config-if)# aps working circuit-number	Configures this interface as a working interface.
Step 3	Router(config-if)# end	Exits configuration mode.
Step 4	Router# show controllers pos	Displays information about the POS controllers and interface so that you can verify that the interface is configured correctly.
	Example: Router# show interfaces pos	
	Example: Router# show aps	

Configuring APS Working and Protect Interfaces**Note**

If a router has two or more protect interfaces, the **aps group** command for each interface must precede the corresponding **aps protect** command.

To configure the protect interface, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. Router(config)# **interface pos slot / port-adapter / port**
2. Router(config-if)# **aps protect circuit-number ip-address**
3. Router(config-if)# **end**
4. Router# **show controllers pos**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config)# interface pos <i>slot / port-adapter / port</i>	Specifies the POS interface to be configured as the protect interface and enters interface configuration mode.
Step 2	Router(config-if)# aps protect <i>circuit-number ip-address</i>	Configures this interface as a protect interface. Specifies the IP address of the router that contains the working interface.
Step 3	Router(config-if)# end	Exits configuration mode.
Step 4	Router# show controllers pos	Displays information about the POS controllers and interface so that you can verify that the interface is configured correctly.
	Example: Router# show interfaces pos	
	Example: Router# show aps	

Configuring Other APS Options

To configure the other APS options, use any of the following optional commands in interface configuration mode.

Command	Purpose
Router(config-if)# aps authenticate <i>string</i>	(Optional) Enables authentication and specifies the string that must be present to accept any packet on the OOB communication channel.
Router(config-if)# aps force <i>circuit-number</i>	(Optional) Manually switches the specified circuit to a protect interface, unless a request of equal or higher priority is in effect.
Router(config-if)# aps group <i>group-number</i>	(Optional) Allows more than one protect/working interface group to be supported on a router.
Router(config-if)# aps lockdown <i>circuit-number</i>	(Optional) Prevents a working interface from switching to a protect interface.
Router(config-if)# aps manual <i>circuit-number</i>	(Optional) Manually switches a circuit to a protect interface, unless a request of equal or higher priority is in effect.
Router(config-if)# aps revert <i>minutes</i>	(Optional) Enables automatic switchover from the protect interface to the working interface after the working interface becomes available.

Command	Purpose
Router(config-if)# aps timers <i>seconds1</i> <i>seconds2</i>	(Optional) Changes the time between hello packets and the time before the protect interface process declares a working interface's router to be down (that is, seconds1 for the hello time and seconds2 for the hold time).
Router(config-if)# aps unidirectional	(Optional) Configures a protect interface for unidirectional mode.

Monitoring and Maintaining APS

To provide information about system processes, the Cisco IOS software includes an extensive list of EXEC commands that begin with the word **show**, which, when executed, display detailed tables of system information. Following is a list of some of the common **show** commands for the APS feature.

To display the information described, use these commands in privileged EXEC mode.

Command	Purpose
Router# show aps	Displays information about the automatic protection switching feature.
Router# show controllers pos	Displays information about the hardware.
Router# show interfaces pos	Displays information about the interface.

Configuring SONET Alarm Reporting

To configure the thresholds and the type of SONET alarms that are reported, use any of the following commands in interface configuration mode. The commands listed in this section are optional. The default settings are adequate for most POS installations.

Command	Purpose
Router(config-if)# pos threshold { b1-tca b2-tca b3-tca sd-ber sf-ber } <i>rate</i>	(Optional) Configures the BER threshold values for signal failure (SF), signal degrade (SD), or threshold crossing alarms (TCAs).
Router(config-if)# pos report { b1-tca b2-tca b3-tca lais lrldi pais plop prdi rdool sd-ber sf-ber slof slos }	(Optional) Enables reporting of selected SONET alarms.

To display the current BER threshold setting or to view the reporting of the SONET alarms, use the **show controllers pos** EXEC command.

Configuring a Protection Switch

LAIS can be used to force a protection switch in an APS environment. To force an APS switch when the interface is placed in administrative shut down state, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# pos ais-shut</code>	Sends line alarm indication signals.

Configuring Serial Interfaces for CSU DSU Service Modules

The Cisco T1 data service unit/channel service unit (DSU/CSU) WAN interface card is an integrated, managed T1 or fractional T1 WAN interface card. It provides nonchannelized data rates of 1 to 24 X 64 kbps or 1 to 24 X 56 kbps and follows ANSI T1.403 and AT&T Publication 62411 standards.

The Cisco DSU/CSU WAN T1 interface includes the following management features:

- You can remotely configure the interface using Telnet and the Cisco IOS command-line interface (CLI).
- For monitoring purposes, the router and DSU/CSU are manageable as a single Simple Network Management Protocol (SNMP) entity using CiscoWorks or CiscoView. DSU/CSU statistics are accessed from the CLI.
- The SNMP agent supports the standard MIB II, Cisco integrated DSU/CSU MIB, and T1 MIB (RFC 1406).
- Loopbacks (including a manual button for a network line loopback) and bit error rate tester (BERT) tests are provided for troubleshooting.
- Test patterns, alarm counters, and performance reports are accessible using the CLI.
- The module has carrier detect, loopback, and alarm LEDs.

The following CSU and DSU service modules are described in this section:

- Fractional T1/FT/WIC CSU/DSU service module
- 2-wire and 4-wire, 56/64-kbps CSU/DSU service module
- [Fractional T1 FT WIC CSU DSU Service Module Configuration Task List, page 119](#)
- [2-Wire and 4-Wire 56 64-kbps CSU DSU Service Module Configuration Task List, page 124](#)

Fractional T1 FT WIC CSU DSU Service Module Configuration Task List

To configure fractional T1 and T1 (FT1/T1) service modules, perform the tasks described in these sections. Each task in the list is identified as either required or optional.

- [Specifying the Clock Source, page 120](#)
- [Enabling Data Inversion Before Transmission, page 120](#)
- [Specifying the Frame Type of an FT T1 Line, page 120](#)
- [Specifying the CSU Line Build-Out, page 121](#)
- [Specifying FT1 T1 Line-Code Type, page 121](#)
- [Enabling Remote Alarms, page 121](#)
- [Enabling Loop Codes That Initiate Remote Loopbacks, page 122](#)
- [Specifying Time Slots, page 123](#)
- [Enabling the T1 CSU WIC, page 123](#)

Specifying the Clock Source

To specify the clock source (that is, the source of the timing synchronization signal) for the FT1/T1 CSU/DSU module, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# service-module t1 clock source {internal line}	Specifies the clock source, for the CSU/DSU internal clock or the line clock.

Enabling Data Inversion Before Transmission

Data inversion is used to guarantee the ones density requirement on an alternate mark inversion (AMI) line when using bit-oriented protocols such as High-Level Data Link Control (HDLC), PPP, X.25, and Frame Relay.

To guarantee the ones density requirement on an AMI line using the FT1/T1 CSU/DSU module, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# service-module t1 data-coding inverted	Inverts bit codes by changing all 1 bits to 0 bits and all 0 bits to 1 bits.

If the time-slot speed is set to 56 kbps, this command is rejected because line density is guaranteed when transmitting at 56 kbps. Use this command with the 64-kbps line speed. If you transmit inverted bit codes, both CSU/DSUs must have this command configured for successful communication.

To enable normal data transmission on an FT1/T1 network, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# service-module tx1 data-coding normal	Enables normal data transmission on a T1 network.
or	
Router(config-if)# no service-module t1 data-coding inverted	

Specifying the Frame Type of an FT T1 Line

To specify the frame type for a line using the FT1/T1 CSU/DSU module, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# service-module t1 framing {sf esf}</code>	Specifies a FT1/T1 frame type. Choose either D4 Super Frame (sf) or Extended Super Frame (esf).

In most cases, the service provider determines which framing type, either **sf** or **esf**, is required for your circuit.

Specifying the CSU Line Build-Out

To decrease the outgoing signal strength to an optimum value for the telecommunication carrier network, use the following command on the FT1/T1 CSU/DSU module in interface configuration mode.

Command	Purpose
<code>Router(config-if)# service-module t1 lbo {-15 db -7.5 db}</code>	Decreases the outgoing signal strength in decibels.

To transmit packets without decreasing outgoing signal strength, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# service-module t1 lbo none</code>	Transmits packets without decreasing outgoing signal strength.

The ideal signal strength should be between -15 dB and -22 dB, which is calculated by adding the phone company loss plus cable length loss plus line build out. You may use this command in back-to-back configurations, but it is not needed on most actual T1 lines.

Specifying FT1 T1 Line-Code Type

To configure the line code for the FT1/T1 CSU/DSU module, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# service-module t1 linecode {ami b8zs}</code>	Specifies a line-code type. Choose alternate mark inversion (AMI) or binary 8 zero substitution (B8ZS).

Configuring B8ZS is a method of ensuring the ones density requirement on a T1 line by substituting intentional bipolar violations in bit positions four and seven for a sequence of eight zero bits. When you configure the CSU/DSU AMI, you must guarantee the ones density requirement in your router using the **service-module t1 data-coding inverted** command or the **service-module t1 timeslots speed 56** command. In most cases, your T1 service provider determines which line-code type, either **ami** or **b8zs**, is required for your T1 circuit.

Enabling Remote Alarms

To generate remote alarms (yellow alarms) at the local CSU/DSU or to detect remote alarms sent from the remote CSU/DSU, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# service-module t1 remote-alarm-enable</code>	Enables remote alarms.

Remote alarms are transmitted by the CSU/DSU when it detects an alarm condition, such as a red alarm (loss of signal) or blue alarm (unframed 1s). The receiving CSU/DSU then knows that there is an error condition on the line.

With D4 Superframe configured, a remote alarm condition is transmitted by setting the bit 2 of each time slot to zero. For received user data that has bit 2 of each time slot set to zero, the CSU/DSU interprets the data as a remote alarm and interrupts data transmission, which explains why remote alarms are disabled by default. With Extended Super Frame configured, the remote alarm condition is signalled out of band in the facility data link.

You can see if the FT1/T1 CSU/DSU is receiving a remote alarm (yellow alarm) by issuing the **show service-module** command.

To disable remote alarms, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# no service-module t1 remote-alarm-enable</code>	Disables remote alarms.

Enabling Loop Codes That Initiate Remote Loopbacks

To specify if the fractional T1/T1 CSU/DSU module goes into loopback when it receives a loopback code on the line, use the following commands in interface configuration mode.

SUMMARY STEPS

1. Router(config-if)# **service-module t1 remote-loopback full**
2. Router(config-if)# **service-module t1 remote-loopback payload [alternate | v54]**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config-if)# service-module t1 remote-loopback full	Configures the remote loopback code used to transmit or accept CSU loopback requests.
Step 2	Router(config-if)# service-module t1 remote-loopback payload [alternate v54]	Configures the loopback code used by the local CSU/DSU to generate or detect payload-loopback commands.



Note

By using the **service-module t1 remote-loopback** command without specifying any keywords, you enable the standard-loopup codes, which use a 1-in-5 pattern for loopup and a 1-in-3 pattern for loopdown.

You can simultaneously configure the **full** and **payload** loopback points. However, only one loopback payload code can be configured at a time. For example, if you configure the **service-module t1 remote-**

loopback payload alternate command, a payload v.54 request, which is the industry standard and default, cannot be transmitted or accepted. Full and payload loopbacks with standard-loopup codes are enabled by default.

The **no** form of this command disables loopback requests. For example, the **no service-module t1 remote-loopback full** command ignores all full-bandwidth loopback transmissions and requests. Configuring the **no** form of the command may not prevent telco line providers from looping your router in **esf** mode, because fractional T1/T1 telcos use facilities data-link messages to initiate loopbacks.

If you enable the **service-module t1 remote-loopback** command, the **loopback remote** commands on the FT1/T1 CSU/DSU module will not be successful.

Specifying Time Slots

To define time slots for an FT1/T1 module, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# service-module t1 timeslots { <i>range</i> all } [speed { 56 64 }]	Specifies time slots.

This command specifies which time slots are used in fractional T1 operation and determines the amount of bandwidth available to the router in each time slot. The *range* specifies the DS0 time slots that constitute the FT1/T1 channel. The range is from 1 to 24, where the first time slot is numbered 1, and the last time slot is numbered 24. Specify this field by using a series of subranges separated by commas. The time-slot range must match the time slots assigned to the channel group. In most cases, the service provider defines the time slots that comprise a channel group. Use the **no** form of this command to select all FT1/T1 time slots that are transmitting at 64 kbps, which is the default.

To use the entire T1 line, enable the **service-module t1 timeslots all** command.

Enabling the T1 CSU WIC

The following are prerequisites to enable the T1 CSU WIC:

- Leased line from your telephone company
- Configuration parameters depending on your specific telephone company. For most connections, the default settings should suffice:
 - **service-module t1 clock source line**
 - **service-module t1 data-coding normal**
 - **service-module t1 timeslots all speed 64**
 - **service-module t1 framing esf**
 - **service-module t1 lbo none**
 - **service-module t1 linecode b8zs**
 - **no service-module t1 remote-alarm-enable**
 - **no service-module t1 fdl**

**Note**

To view the current configuration, use the **show service-module serial slot / port** command. For further information about these commands and how to change them, refer to the Cisco IOS configuration guides and command references that shipped with your router.

To configure the router to send SNMP traps, use the following commands:

SUMMARY STEPS

1. Router(config)# **interface serial slot / port**
2. Router(config-if)# **service-module t1 fdl {ansi | att}**
3. Router(config-if)# **exit**
4. Router(config)# **more system:running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config)# interface serial slot / port	Enters interface configuration mode. The <i>slot / port</i> argument corresponds to where the WAN interface card is installed in your router.
Step 2	Router(config-if)# service-module t1 fdl {ansi att}	Sets the fdl parameter to either ansi or att .
Step 3	Router(config-if)# exit	Exits interface configuration mode.
Step 4	Router(config)# more system:running-config	Displays the fdl parameter so that you can verify that it has changed.

2-Wire and 4-Wire 56 64-kbps CSU DSU Service Module Configuration Task List

To configure 2- and 4-wire, 56/64 kbps service modules, perform the tasks described in these sections:

- [Setting the Clock Source, page 124](#)
- [Setting the Network Line Speed, page 125](#)
- [Enabling Scrambled Data Coding, page 125](#)
- [Changing Between Digital Data Service and Switched Dial-Up Modes, page 126](#)
- [Enabling Acceptance of a Remote Loopback Request, page 126](#)
- [Selecting a Service Provider, page 127](#)

Setting the Clock Source

In most applications, the CSU/DSU should be configured using the **service-module 56k clock source line** command. For back-to-back configurations, use the **internal** keyword to configure one CSU/DSU and use the **line** keyword to configure the other CSU/DSU.

To configure the clock source for a 4-wire, 56/64-kbps CSU/DSU module, use the following command in interface configuration mode for a serial interface:

Command	Purpose
<code>Router(config-if)# service-module 56k clock source {line internal}</code>	Configures the clock source.

Use the **no** form of this command to revert to the default clock source, which is the line clock.

Setting the Network Line Speed

To configure the network line speed for a 4-wire, 56/64-kbps CSU/DSU module, use the following command in interface configuration mode for a serial interface:

Command	Purpose
<code>Router(config-if)# service-module 56k clock rate speed</code>	Sets the network line speed.

You can use the following line speed settings: 2.4, 4.8, 9.6, 19.2, 38.4, 56, 64 kbps, and an **auto** setting.

The 64-kbps line speed cannot be used with back-to-back digital data service (DDS) lines. The substrate line speeds are determined by the service provider.

Only the 56-kbps line speed is available in switched mode. Switched mode is the default on the 2-wire CSU/DSU and is enabled by the **service-module 56k network-type** interface configuration command on the 4-wire CSU/DSU.

The **auto** linespeed setting enables the CSU/DSU to decipher current line speed from the sealing current running on the network. Because back-to-back DDS lines do not have sealing current, use the **auto** setting only when transmitting over telco DDS lines and using the line clock as the clock source.

Use the **no** form of this command to enable a network line speed of 56 kbps, which is the default.

Enabling Scrambled Data Coding

To prevent application data from replicating loopback codes when operating at 64 kbps on a 4-wire CSU/DSU, use the following command in interface configuration mode for a serial interface:

Command	Purpose
<code>Router(config-if)# service-module 56k data-coding scrambled</code>	Scrambles bit codes before transmission.

Enable the scrambled configuration only in 64 kbps DDS mode. If the network type is set to switched, the configuration is refused.

If you transmit scrambled bit codes, both CSU/DSUs must have this command configured for successful communication.

To enable normal data transmission for the 4-wire, 56/64-kbps module, use one of the following commands for a serial interface in interface configuration mode.

Command	Purpose
Router(config-if)# service-module 56k data-coding normal	Specifies normal data transmission.
or	
Router(config-if)# no service-module 56k data-coding	

Changing Between Digital Data Service and Switched Dial-Up Modes

To transmit packets in DDS mode or switched dial-up mode using the 4-wire, 56/64-kbps CSU/DSU module, use one of the following commands in interface configuration mode for a serial interface:

Command	Purpose
Router(config-if)# service-module 56k network-type dds	Transmits packets in DDS mode or switched dial-up mode.
or	
Router(config-if)# service-module 56k network-type switched	

Use the **no** form of these commands to transmit from a dedicated leased line in DDS mode. DDS mode is enabled by default for the 4-wire CSU/DSU. Switched mode is enabled by default for the 2-wire CSU/DSU.

In switched mode, you need additional dialer configuration commands to configure dial-out numbers. Before you enable the **service-module 56k network-type switched** command, both CSU/DSUs must use a clock source coming from the line and the clock rate must be configured to **auto** or **56k**. If the clock rate is not set correctly, this command will not be accepted.

The 2-wire and 4-wire, 56/64-kbps CSU/DSU modules use V.25 *bis* dial commands to interface with the router. Therefore, the interface must be configured using the **dialer in-band** command. DTR dial is not supported.



Note

Any loopbacks in progress are terminated when switching between modes.

Enabling Acceptance of a Remote Loopback Request

To enable the acceptance of a remote loopback request on a 2- or 4-wire, 56/64-kbps CSU/DSU module, use the following command in interface configuration mode for a serial interface:

Command	Purpose
<code>Router(config-if)# service-module 56k remote-loopback</code>	Enables a remote loopback request.

The **no service-module 56k remote-loopback** command prevents the local CSU/DSU from being placed into loopback by remote devices on the line. Unlike the T1 module, the 2- or 4-wire, 56/64-kbps CSU/DSU module can still initiate remote loopbacks with the **no** form of this command configured.

Selecting a Service Provider

To select a service provider to use with a 2- or 4-wire, 56/64 kbps dial-up line, use the following command in interface configuration mode for a serial interface:

Command	Purpose
<code>Router(config-if)# service-module 56k switched-carrier {att other sprint}</code>	Selects a service provider for a 2- or 4-wire switched, 56/64 kbps dialup line.

The **att** keyword specifies AT&T or another digital network service provider as the line carrier, which is the default for the 4-wire, 56/64-kbps CSU/DSU module. The **sprint** keyword specifies Sprint or another service provider whose network carries mixed voice and data as the line carrier, which is the default for the 2-wire switched 56-kbps CSU/DSU module.

In a Sprint network, echo-canceler tones are sent during call setup to prevent echo cancelers from damaging digital data. The transmission of these cancelers may increase call setup times by 8 seconds on the 4-wire module. Having echo cancellation enabled does not affect data traffic.

This configuration command is ignored if the network type is DDS.

Use the **no** form of this command to enable the default service provider. AT&T is enabled by default on the 4-wire, 56/64 module. Sprint is enabled by default on the 2-wire switched, 56-kbps module.

Configuring Low-Speed Serial Interfaces

This section describes how to configure low-speed serial interfaces. In addition to the background information described in the [Understanding Half-Duplex DTE and DCE State Machines, page 127](#), these sections provide guidelines for configuring low-speed serial interfaces:

For configuration examples, see the [Low-Speed Serial Interface Examples, page 155](#).

- [Understanding Half-Duplex DTE and DCE State Machines, page 127](#)
- [Changing Between Controlled-Carrier and Constant-Carrier Modes, page 131](#)
- [Tuning Half-Duplex Timers, page 132](#)
- [Changing Between Synchronous and Asynchronous Modes, page 133](#)

Understanding Half-Duplex DTE and DCE State Machines

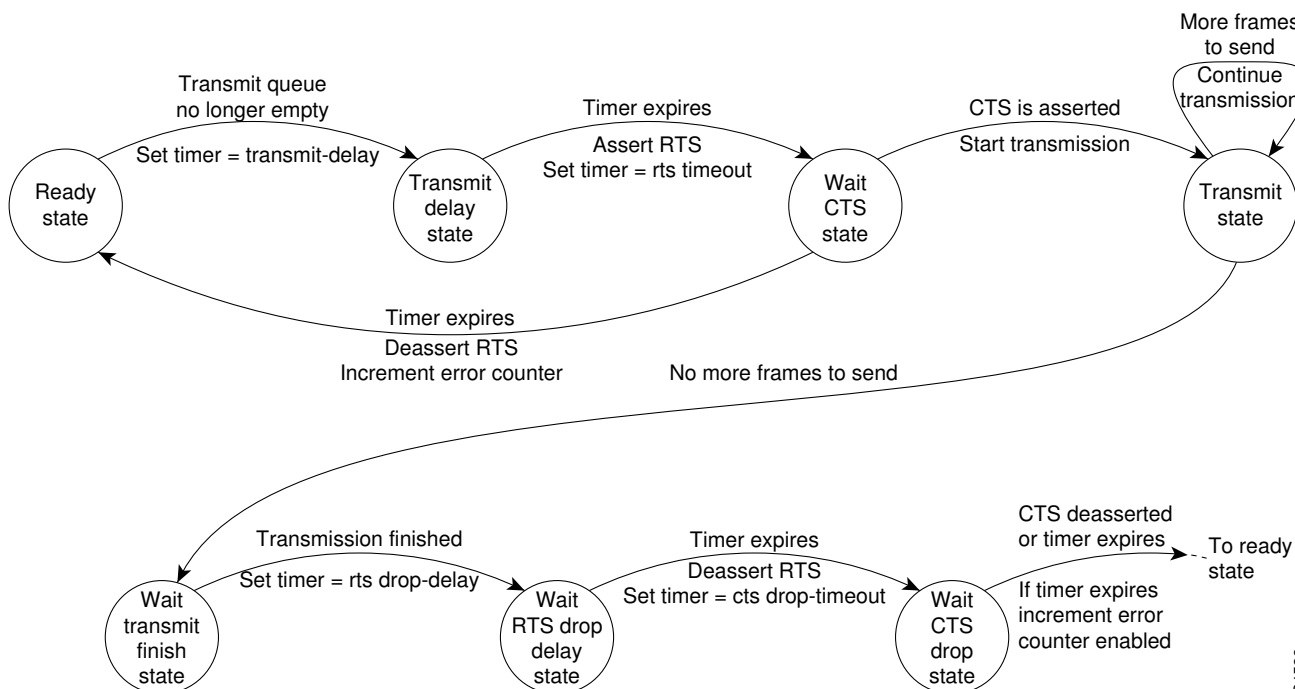
The following sections describe the communication between half-duplex DTE transmit and receive state machines and half-duplex DCE transmit and receive state machines.

- [Half-Duplex DTE State Machines, page 128](#)
- [Half-Duplex DCE State Machines, page 129](#)

Half-Duplex DTE State Machines

As shown in the figure below, the half-duplex DTE transmit state machine for low-speed interfaces remains in the ready state when it is quiescent. When a frame is available for transmission, the state machine enters the transmit delay state and waits for a time period, which is defined by the **half-duplex timer transmit-delay** command. The default is 0 milliseconds. Transmission delays are used for debugging half-duplex links and assisting lower-speed receivers that cannot process back-to-back frames.

Figure 4 Half-Duplex DTE Transmit State Machine



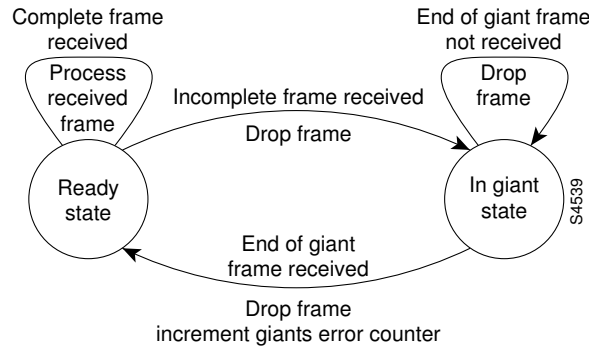
After idling for a defined number of milliseconds (ms), the state machine asserts a request to send (RTS) signal and changes to the wait-clear-to-send (CTS) state for the DCE to assert CTS. A timeout timer with a value set by the **half-duplex timer rts-timeout** command starts. This default is 3 ms. If the timeout timer expires before CTS is asserted, the state machine returns to the ready state and deasserts RTS. If CTS is asserted before the timer expires, the state machine enters the transmit state and sends the frames.

Once there are no more frames to transmit, the state machine transitions to the wait transmit finish state. The machine waits for the transmit FIFO in the serial controller to empty, starts a delay timer with a value defined by the **half-duplex timer rts-drop-delay** interface command, and transitions to the wait RTS drop delay state.

When the timer in the wait RTS drop delay state expires, the state machine deasserts RTS and transitions to the wait CTS drop state. A timeout timer with a value set by the **half-duplex timer cts-drop-timeout** interface command starts, and the state machine waits for the CTS to deassert. The default is 250 ms. Once the CTS signal is deasserted or the timeout timer expires, the state machine transitions back to the ready state. If the timer expires before CTS is deasserted, an error counter is incremented, which can be displayed by issuing the **show controllers** command for the serial interface in question.

As shown in the figure below, a half-duplex DTE receive state machine for low-speed interfaces idles and receives frames in the ready state. A giant frame is any frame whose size exceeds the maximum transmission unit (MTU). If the beginning of a giant frame is received, the state machine transitions to the in giant state and discards frame fragments until it receives the end of the giant frame. At this point, the state machine transitions back to the ready state and waits for the next frame to arrive.

Figure 5 Half-Duplex DTE Receive State Machine



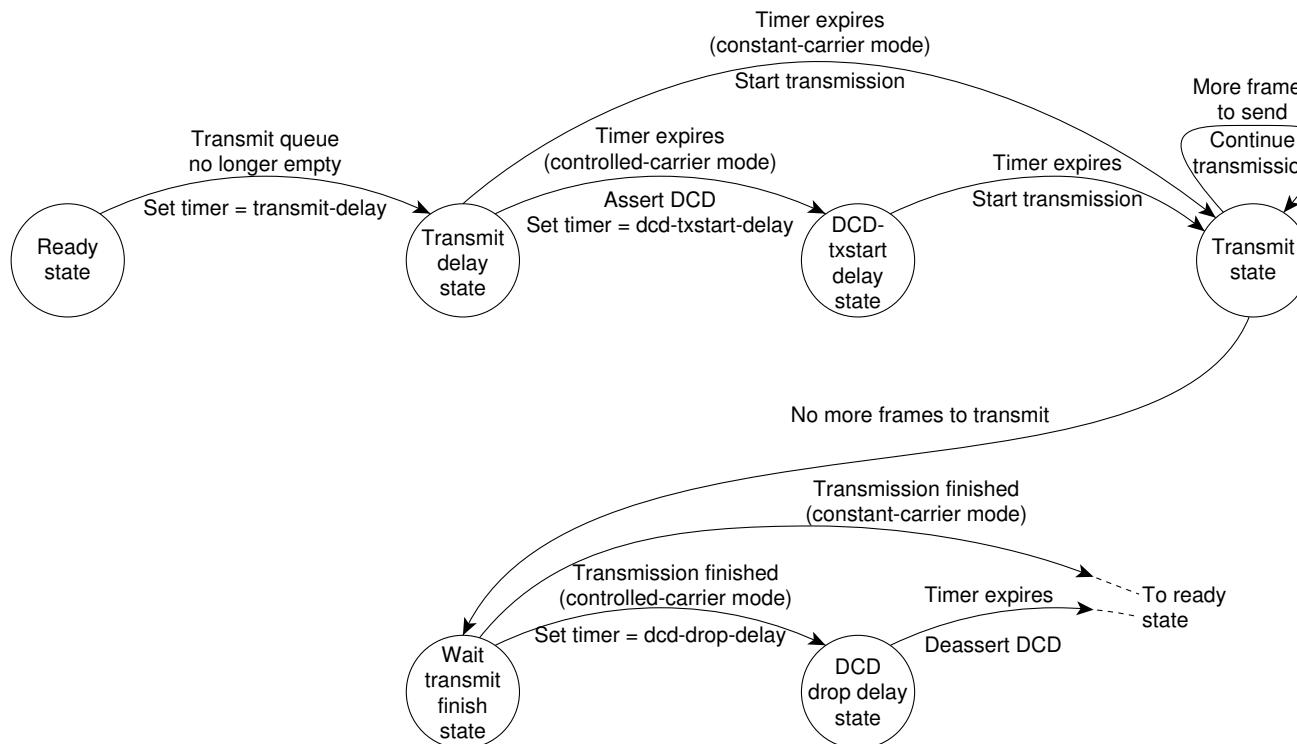
An error counter is incremented upon receipt of the giant frames. To view the error counter, use the **show interfaces** command for the serial interface in question.

Half-Duplex DCE State Machines

As shown in the figure below, for a low-speed serial interface in DCE mode, the half-duplex DCE transmit state machine idles in the ready state when it is quiescent. When a frame is available for transmission on the serial interface, such as when the output queues are no longer empty, the state machine starts a timer (based on the value of the **half-duplex timer transmit-delay** command, in milliseconds) and transitions to the transmit delay state. Similar to the DTE transmit state machine, the transmit delay state gives you the option of setting a delay between the transmission of frames; for example, this feature lets you compensate for a slow receiver that loses data when multiple frames are received in quick succession. The default

transmit-delay value is 0 ms; use the **half-duplex timer transmit-delay** interface configuration command to specify a delay value not equal to 0.

Figure 6 Half-Duplex DCE Transmit State Machine



After the transmit delay state, the next state depends on whether the interface is in constant-carrier mode (the default) or controlled-carrier mode.

If the interface is in constant-carrier mode, it passes through the following states:

- 1 The state machine passes to the transmit state when the **transmit-delay** timer expires. The state machine stays in the transmit state until there are no more frames to transmit.
- 2 When there are no more frames to transmit, the state machine passes to the wait transmit finish state, where it waits for the transmit FIFO to empty.
- 3 Once the FIFO empties, the DCE passes back to the ready state and waits for the next frame to appear in the output queue.

If the interface is in controlled-carrier mode, the interface performs a handshake using the data carrier detect (DCD) signal. In this mode, DCD is deasserted when the interface is idle and has nothing to transmit. The transmit state machine transitions through the states as follows:

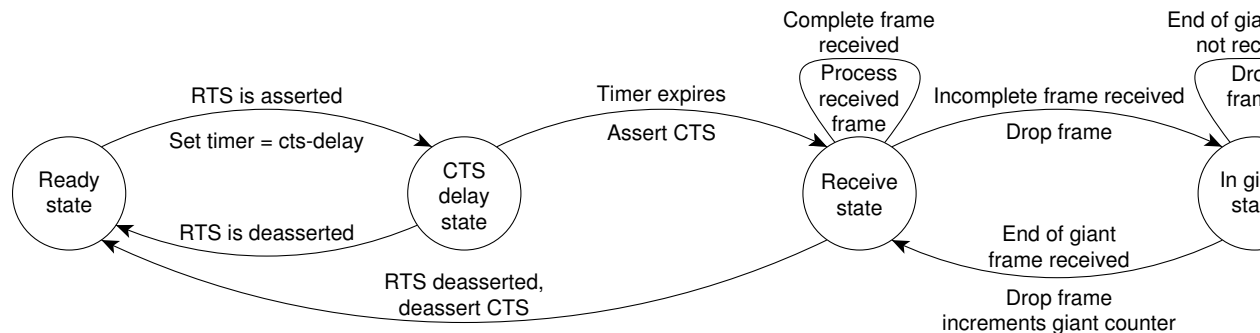
- 1 After the **transmit-delay** timer expires, the DCE asserts DCD and transitions to the DCD-txstart delay state to ensure a time delay between the assertion of DCD and the start of transmission. A timer is started based on the value specified using the **dcd-txstart-delay** command. (This timer has a default value of 100 ms; use the **half-duplex timer dcd-txstart-delay** interface configuration command to specify a delay value.)
- 2 When this delay timer expires, the state machine transitions to the transmit state and transmits frames until there are no more frames to transmit.
- 3 After the DCE transmits the last frame, it transitions to the wait transmit finish state, where it waits for transmit FIFO to empty and the last frame to transmit to the wire. Then DCE starts a delay timer by

specifying the value using the **dcd-drop-delay** command. (This timer has the default value of 100 ms; use the **half-duplex timer dcd-drop-delay** interface configuration command to specify a delay value.)

- 4 The DCE transitions to the wait DCD drop delay state. This state causes a time delay between the transmission of the last frame and the deassertion of DCD in the controlled-carrier mode for DCE transmits.
- 5 When the timer expires, the DCE deasserts DCD and transitions back to the ready state and stays there until there is a frame to transmit on that interface.

As shown in the figure below, the half-duplex DCE receive state machine idles in the ready state when it is quiescent. It transitions out of this state when the DTE asserts RTS. In response, the DCE starts a timer based on the value specified using the **cts-delay** command. This timer delays the assertion of CTS because some DTE interfaces expect this delay. (The default value of this timer is 0 ms; use the **half-duplex timer cts-delay** interface configuration command to specify a delay value.)

Figure 7 Half-Duplex DCE Receive State Machine



When the timer expires, the DCE state machine asserts CTS and transitions to the receive state. It stays in the receive state until there is a frame to receive. If the beginning of a giant frame is received, it transitions to the in giant state and keeps discarding all the fragments of the giant frame and transitions back to the receive state.

Transitions back to the ready state occur when RTS is deasserted by the DTE. The response of the DCE to the deassertion of RTS is to deassert CTS and go back to the ready state.

Changing Between Controlled-Carrier and Constant-Carrier Modes

The **half-duplex controlled-carrier** command enables you to change between controlled-carrier and constant-carrier modes for low-speed serial DCE interfaces in half-duplex mode. Configure a serial interface for half-duplex mode by using the **half-duplex** command. Full-duplex mode is the default for serial interfaces. This interface configuration is available on Cisco 2520 through Cisco 2523 routers.

Controlled-carrier operation means that the DCE interface will have DCD deasserted in the quiescent state. When the interface has something to transmit, it will assert DCD, wait a user-configured amount of time, then start the transmission. When it has finished transmitting, it will again wait a user-configured amount of time and then deassert DCD.

- [Placing a Low-Speed Serial Interface in Controlled-Carrier Mode, page 131](#)
- [Placing a Low-Speed Serial Interface in Constant-Carrier Mode, page 132](#)

Placing a Low-Speed Serial Interface in Controlled-Carrier Mode

To place a low-speed serial interface in controlled-carrier mode, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# half-duplex controlled-carrier	Places a low-speed serial interface in controlled-carrier mode.

Placing a Low-Speed Serial Interface in Constant-Carrier Mode

To return a low-speed serial interface to constant-carrier mode from controlled-carrier mode, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# no half-duplex controlled-carrier	Places a low-speed serial interface in constant-carrier mode.

Tuning Half-Duplex Timers

To optimize the performance of half-duplex timers, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# half-duplex timer { cts-delay <i>value</i> cts-drop-timeout <i>value</i> dcd-drop-delay <i>value</i> dcd-txstart-delay <i>value</i> rts-drop-delay <i>value</i> rts-timeout <i>value</i> transmit-delay <i>value</i> } }	Tunes half-duplex timers.

The timer tuning commands permit you to adjust the timing of the half-duplex state machines to suit the particular needs of their half-duplex installation.

Note that the **half-duplex timer** command and its options replaces the following two timer tuning commands that are available only on high-speed serial interfaces:

- **sdhc cts-delay**
- **sdhc rts-timeout**

Changing Between Synchronous and Asynchronous Modes

To specify the mode of a low-speed serial interface as either synchronous or asynchronous, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# physical-layer {sync async}</code>	Specifies the mode of a low-speed interface as either synchronous or asynchronous.

This command applies only to low-speed serial interfaces available on Cisco 2520 through Cisco 2523 routers.



Note

When you make a transition from asynchronous mode to synchronous mode in serial interfaces, the interface state becomes down by default. You should then use **no shutdown** option to bring the interface up.

In synchronous mode, low-speed serial interfaces support all interface configuration commands available for high-speed serial interfaces, except the following two commands:

- **sdlc cts-delay**
- **sdlc rts-timeout**

When placed in asynchronous mode, low-speed serial interfaces support all commands available for standard asynchronous interfaces. The default is synchronous mode.



Note

When you use this command, it does not appear in the output of the **show running-config** and **show startup-config** commands, because the command is a physical-layer command.

To return to the default mode (synchronous) of a low-speed serial interface on a Cisco 2520 through Cisco 2523 router, use the following command in interface configuration mode.

Command	Purpose
<code>Router(config-if)# no physical-layer</code>	Returns the interface to its default mode, which is synchronous.

Automatic Removal of tftp ftp rcp Source Interfaces Configuration

A serial interface is configured using controller configuration. For example:

```
76C(config)#controller t1 2/1/0
76C(config-controller)#channel-group 1 timeslots 1-24
76C(config-controller)#end
```

The configured interface is now configured as tftp or ftp or rcpc source interfaces. For example:

```
76C(config)#ip ftp source-interface Serial2/1/0:1
OR
76C(config)#ip tftp source-interface Serial2/1/0:1
OR
76C(config)#ip rcpc source-interface Serial2/1/0:1
```

Remove the controller configuration:

```
76C(config)#controller t1 2/1/0
76C(config-controller)#no channel-group 1 timeslots 1-24
76C(config-controller)#end
```

Now the tftp or ftp or rcpc source interfaces configured as Serial2/1/0:1 are automatically unconfigured. You need not manually unconfigure them after you remove the configuration for controller interface, which was used as source interface for tftp or ftp or rcpc.

Troubleshooting Serial Interfaces

Perform the tasks in this section to troubleshoot issues with serial interfaces:

- [Troubleshooting Channelized T1 or E1, page 134](#)
- [Troubleshooting the T3 and T1 Channels on the CT3IP, page 137](#)
- [Troubleshooting the PA-E3 Port Adapter, page 142](#)

Troubleshooting Channelized T1 or E1

When troubleshooting channelized T1 or E1, you must first determine if the problem is with a particular channel group or with the T1 or E1 line.

If the problem is with a single channel group, you have a potential interface problem.

If the problem is with the T1 or E1 line, or with all channel groups, you have a potential controller problem.

The following section describes how to determine whether the problem affects an interface or a controller:

When you troubleshoot E1 or T1 controllers, first check that the configuration is correct. The framing type and line code should match what the service provider has specified. Then check channel group and PRI-group configurations, especially to verify that the time slots and speeds are what the service provider has specified.

At this point, the **show controllers t1** or **show controllers e1** commands should be used to check for T1 or E1 errors. Use the command several times to determine if error counters are increasing, or if the line status is continually changing. If these errors are occurring, you need to work with the service provider.



Note

Cisco routers do not have CSU capability and do not react to any remote loopback codes at the T1 or E1 level.

- [Running Controller Loopback Diagnostic Tests, page 134](#)

Running Controller Loopback Diagnostic Tests

Controller loopback tests are a means to isolate problems and are available for both channelized T1 controllers and channelized E1 controllers. The following loopback tests are documented for isolating T1 and E1 controller issues:

- [Local Loopback, page 135](#)
- [Remote Loopback, page 135](#)
- [Channelized E1 Controller Loopback, page 136](#)

Local Loopback

The local loopback loops the controller both toward the router and toward the line. Because the loopback is done internally to the router, the controller should make the transition to the UP state within approximately 10 seconds, and no further T1 errors should be detected.

All channel groups will be looped back; if the encapsulation on that channel group supports loopbacks (for example, HDLC and PPP), you can test that channel group by pinging the interface address. For example, if you have assigned an IP address to the serial interface defined for a channel group, you can ping that IP address.

To place the controller into local loopback, use the following command in controller configuration mode.

Command	Purpose
<code>Router(config-controller)# loopback local controller</code>	Loops the T1 controller toward the router and toward the line.

To test a channel group, use the following command in EXEC mode.

Command	Purpose
<code>Router# ping protocol protocol-address</code>	Pings the interface address.

To check errors, use the following command in EXEC mode.

Command	Purpose
<code>Router> show controllers t1</code>	Checks errors.

If any errors occur, or the controller fails to change to the up state, contact the Cisco Technical Assistance Center (TAC).

Because the controller local loopback is bidirectional, the service provider can test the line integrity using a T1 bit error rate tester (BERT) test set.

Remote Loopback

The second T1 controller loopback is a remote loopback. This loopback can be used only if the *entire* T1 goes to a remote CSU. This is not the case with 99.9 percent of channelized T1. When the **loopback remote controller** command is executed, an in-band CSU loop-up code will be sent over the entire T1, which will attempt to loop up the remote CSU. To place the controller in remote loopback, use the following command in controller configuration mode.

Command	Purpose
Router(config-controller)# loopback remote controller	Places the T1 controller in remote loopback.

**Note**

If controller loopbacks are used, they will disrupt service for all channel groups on that interface.

Channelized E1 Controller Loopback

For the E1 controller, only the local loopback is available. Local loopback operates the same as the local loopback on the T1 controller, forming a bidirectional loopback, both toward the router and toward the line. To place the E1 controller in local loopback, use the following command in controller configuration mode.

Command	Purpose
Router(config-controller)# loopback controller	Places the E1 controller in local loopback toward the router and toward the line.

All channel groups will be looped back; if the encapsulation on that channel group supports loopbacks (for example, HDLC and PPP), you can test that channel group by pinging the interface address. For example, if you have assigned an IP address to the serial interface defined for a channel group, you can ping that IP address.

To place the controller into local loopback, use the following command in controller configuration mode.

Command	Purpose
Router(config-controller)# loopback local controller	Loops the T1 controller toward the router and toward the line.

To test a channel group, use the following command in EXEC mode.

Command	Purpose
Router> ping protocol protocol-address	Pings the interface address.

To check errors, if any, use the following command in EXEC mode.

Command	Purpose
Router> show controllers t1	Checks errors.

If any errors occur, they are most likely a hardware problem; contact the Cisco TAC. In addition, you can ask the service provider to test the line by using a T1 BERT test set.

Troubleshooting the T3 and T1 Channels on the CT3IP

To troubleshoot the CT3IP using Cisco IOS software, use the following methods:

- Test the T1 by using the **t1 testcontroller** configuration command and the test port.
- Loop the T1 by using **loopback** interface configuration commands.
- Loop the T3 by using **loopback** controller configuration commands.

- [Enabling Test Port, page 137](#)
- [Enabling Test Port, page 138](#)

Enabling Test Port

You can use the T1 test port available on the CT3IP to break out any of the 28 T1 channels for testing (for example, 24-hour bit error rate tester (BERT) testing is commonly done by telephone companies before a line is brought into service).

The T1 test port is also available as an external port. For more information on configuring an external port, see the [Configuring External T1 Channels, page 91](#).



Note

If a T1 channel that was previously configured as a serial interface is broken out to the T1 port test, then that interface and its associated configuration remain intact while the channel is broken out to the T1 port test. The serial interface is not usable during the time the T1 channel is broken out to the T1 test port; however, the configuration remains to facilitate the return of the T1 channel to a serial interface using the **no t1 test** command.

To enable a T1 channel as a test port, use the following commands beginning in privileged EXEC mode.

SUMMARY STEPS

1. Router# **show controller t3 slot / port-adapter / port**
2. Router(config)# **controller t3 slot / port-adapter / port**
3. Router(config-controller)# **t1 test channel [cablelength feet] [linecode {ami | b8zs}}**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# show controller t3 slot / port-adapter / port	Displays the Ext1... field so that you can verify whether the external device connected to the external T1 port is configured and cabled correctly. If the line status is OK, a valid signal is being received and the signal is not an all-ones signal.
Step 2	Router(config)# controller t3 slot / port-adapter / port	Selects the CT3IP and enters controller configuration mode.
Step 3	Router(config-controller)# t1 test channel [cablelength feet] [linecode {ami b8zs}}	Enables the T1 channel (values are 1 to 28) as a test port and optionally specifies the cable length and line code. The default cablelength is 133 feet, and the default linecode is b8zs .

Enabling Test Port

To disable a T1 channel as a test port, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. Router(config)# **controller t3** *slot / port-adapter / port*
2. Router(config-controller)# **no t1 test channel**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config)# controller t3 <i>slot / port-adapter / port</i>	Selects the CT3IP and enters controller configuration mode.
Step 2	Router(config-controller)# no t1 test channel	Disables the T1 channel (values are 1 to 28) as a test port.



Note

Although you can specify a cable length from 0 to 655 feet, the hardware only recognizes the following ranges: 0 to 133, 134 to 266, 267 to 399, 400 to 533, and 534 to 655. For example, entering 150 feet uses the 134 to 266 range. If you later change the cable length to 200 feet, there is no change because 200 is within the 134 to 266 range. However, if you change the cable length to 399, the 267 to 399 range is used. The actual number you enter is stored in the configuration file.

- [Loopback T1 Channels, page 138](#)
- [Loopback T3 Lines, page 141](#)

Loopback T1 Channels

You can perform the following types of loopbacks on a T1 channel:

- Local--Loops the router output data back toward the router at the T1 framer and sends an alarm indication signal (AIS) out toward the network (see the first figure below).
- Network line--Loops the data back toward the network before the T1 framer and automatically sets a local loopback (see the second figure below).
- Network payload--Loops just the payload data back toward the network at the T1 framer and automatically sets a local loopback (see the third figure below).
- Remote line inband--Sends a repeating 5-bit inband pattern (00001) to the remote end requesting that it enter into a network line loopback (see the fourth figure below).

To enable loopbacks on a T1 channel, use the first command in global configuration mode, followed by any one of the following commands in interface configuration mode.

SUMMARY STEPS

1. Router(config)# **interface serial** *slot / port-adapter / port:t1-channel*
2. Router(config-if)# **loopback local**

DETAILED STEPS

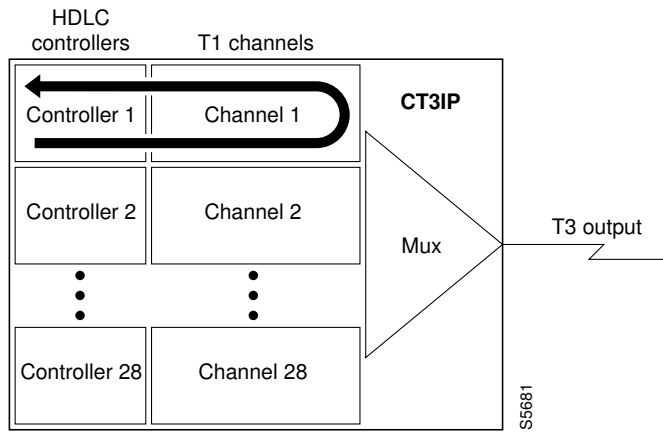
Command or Action	Purpose
Step 1 Router(config)# interface serial slot / port-adapter / port:t1-channel	Selects the T1 channel (values are 1 to 28) on the CT3IP and enters interface configuration mode.
Step 2 Router(config-if)# loopback local Example: Example: Router(config-if)# loopback network line Example: Example: Router(config-if)# loopback network payload Example: Example: Router(config-if)# loopback remote line inband	Enables the local loopback on the T1 channel. Enables the network line loopback on the T1 channel. Enables the network payload loopback on the T1 channel. Enables the remote line inband loopback on the T1 channel.



Note The port adapter and port numbers for the CT3IP are 0.

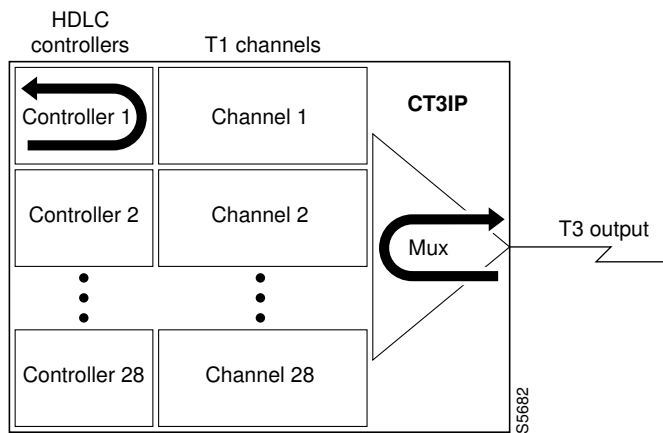
The figure below shows an example of a local loopback in which the loopback occurs in the T1 framer.

Figure 8 *CT3IP Local Loopback*



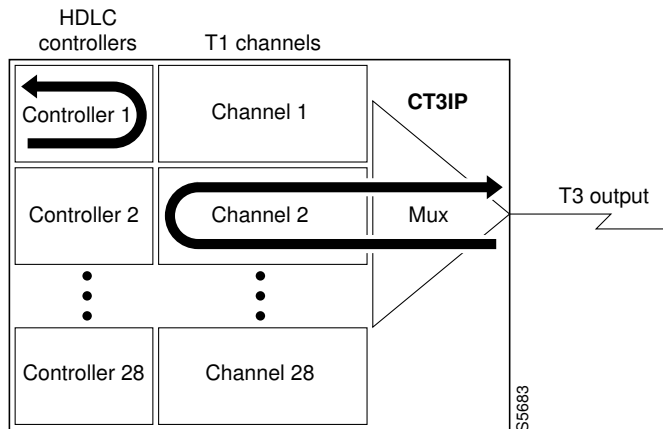
The figure below shows an example of a network line loopback in which just the data is looped back toward the network (before the T1 framer).

Figure 9 *CT3IP Network Line Loopback*



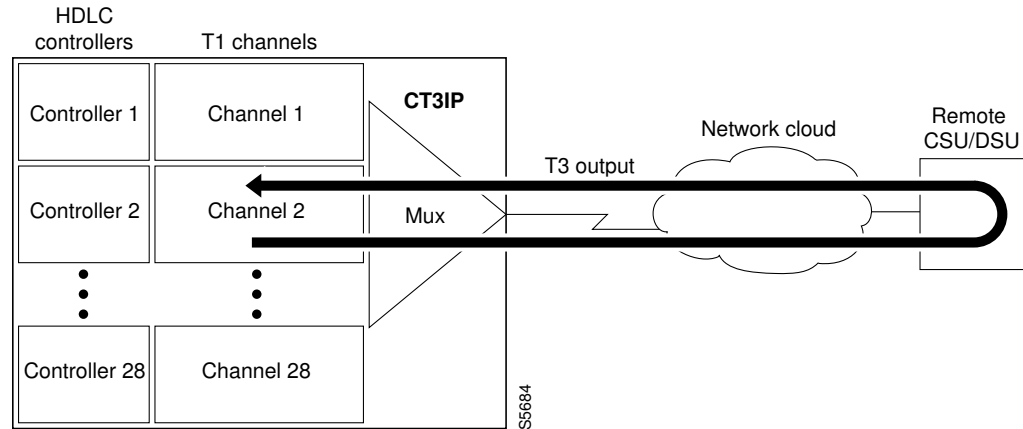
The figure below shows an example of a network payload loopback in which just the payload data is looped back toward the network at the T1 framer.

Figure 10 *CT3IP Network Payload Loopback*



The figure below shows an example of a remote inband loopback in which the network line enters a line loopback.

Figure 11 CT3IP Remote Loopback



Loopback T3 Lines

You can put the entire T3 line into loopback mode (that is, all T1 channels are looped) by using the following types of loopbacks:

- Local--Loops the router output data back toward the router at the T1 framer and sends an AIS signal out toward the network.
- Network--Loops the data back toward the network (before the T1 framer).
- Remote--Sends a FEAC (far-end alarm control) request to the remote end requesting that it enter into a network line loopback. FEAC requests (and therefore remote loopbacks) are possible only when the T3 is configured for C-bit framing. The type of framing used is determined by the equipment to which you are connected. (For more information, refer to the **framing** controller configuration command in the *Cisco IOS Interface and Hardware Component Command Reference* .)

To enable loopbacks on the T3 (and all T1 channels), use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. Router(config)# **controller t3slot/port-adapter/port**
2. Router(config-controller)# **loopback local**

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 Router(config)# controller t3slot/port-adapter/port</p> <p>Example:</p>	<p>Selects the CT3IP and enters controller configuration mode. The port adapter and port numbers for the CT3IP are 0.</p>

Command or Action	Purpose
Step 2 Router(config-controller)# loopback local Example: Router(config-controller)# loopback network Example: Router(config-controller)# loopback remote	(Optional) Enables the local loopback. (Optional) Enables the network loopback. (Optional) Enables the remote loopback.

Troubleshooting the PA-E3 Port Adapter

To set the following loopbacks to troubleshoot the PA-E3 port adapter using Cisco IOS software, use the first command in global configuration mode, followed by any of the other commands, depending on your needs:

Command	Purpose
Router(config)# loopback dte	Loops back after the line interface unit (LIU) toward the terminal.
Router(config)# loopback local	Loops back after going through the framer toward the terminal.
Router(config)# loopback network line	Loops back toward the network before going through the framer.
Router(config)# loopback network payload	Loops back toward the network after going through the framer.

These loopback commands loop all packets from the E3 interface back to the interface and also direct the packets to the network.

Configuration Examples for Serial Interface Configuration

- [Interface Enablement Configuration Examples, page 143](#)
- [HSSI Configuration Examples, page 143](#)
- [Channelized T3 Interface Processor Configuration Examples, page 143](#)
- [PA-E3 Serial Port Adapter Configuration Example, page 146](#)
- [PA-T3 and PA-2T3 Configuration Example, page 147](#)
- [Packet OC-3 Interface Configuration Examples, page 147](#)
- [DPT OC-12c Interface Configuration Examples, page 148](#)
- [APS Configuration Examples, page 149](#)
- [CSU DSU Service Module Examples, page 151](#)

- [Low-Speed Serial Interface Examples, page 155](#)

Interface Enablement Configuration Examples

The following example illustrates how to begin interface configuration on a serial interface. It assigns PPP encapsulation to serial interface 0.

```
interface serial 0
 encapsulation ppp
```

The same example on a Cisco 7500series router, assigning PPP encapsulation to port 0 in slot 1, requires the following commands:

```
interface serial 1/0
 encapsulation ppp
```

This example shows how to configure the access server so that it will use the default address pool on all interfaces except interface 7, on which it will use an address pool called lass:

```
ip address-pool local
ip local-pool lass 172.30.0.1
 async interface
 interface 7
 peer default ip address lass
```

HSSI Configuration Examples

The following example shows a simple configuration for a HSSI port adapter on a Cisco 7500 series router:

```
interface hssi 2/0/0
 ip address 10.1.1.10 255.255.255.0
 description To San Jose, circuit ID 1234
 no ip mroute-cache
```

The following example shows how to configure a 1-port HSSI network module on a Cisco 3600 series router. Both sides of the network connection need to be configured:

```
interface hssi 0/0
 ! Specifies a HSSI interface; changes from global configuration mode to interface
 configuration mode.
 ip address 10.1.1.1 255.255.255.0
 ! Assigns IP address 10.1.1.1 to the interface.
 hssi internal-clock
 ! Converts the HSSI interface into a clock master.
 no fair-queue
 ! Disables weighted fair queueing (WFQ).
 no shutdown
 ! Enables the port.
interface hssi 1/0
 ip address 10.1.1.2 255.255.255.0
 hssi internal-clock
 no fair-queue
 no shutdown
```

Channelized T3 Interface Processor Configuration Examples

The examples in this section show how to configure the Channelized T3 Interface Processor (CT3IP). The first example shows how to configure two of the T1 channels of the channelized T3 controller. The second example shows how to configure one of the T1 channels of the channelized T3 controller as an external port for further channelization on the Multichannel Interface Processor (MIP).

For more information, see the [Configuring T3 Controller Support for the Cisco AS5800](#), page 86, the [Configuring the T3 Controller](#), page 88, and the [Configuring External T1 Channels](#), page 91. The following examples are included in this section:

- [Typical CT3IP Controller Configuration Examples](#), page 144
- [CT3IP Configuration with Default Values Accepted Example](#), page 145
- [CT3IP External Ports Configuration Example](#), page 145
- [CT3IP Maintenance Data Link Example](#), page 145
- [CT3IP Performance Monitoring Example](#), page 145
- [BERT Profile Configuration Example](#), page 146
- [E2 Clock Rate Configuration Example](#), page 146
- [CT3IP BERT Test Pattern Example](#), page 146
- [CT3IP Remote FDL Loopback Example](#), page 146

Typical CT3IP Controller Configuration Examples

A typical T3 controller configuration in a running-configuration file follows:

T3 controller configuration:

```
-----
controller T3 1/0/0
 framing m23
 clock source line
 cablelength 224
 t1 1 controller
 t1 2 controller
 t1 3 controller
 t1 4 controller
 t1 5 controller
 t1 6 controller
 t1 7 controller
 t1 8 controller
 t1 9 controller
 t1 10 controller
 t1 11 controller
 t1 12 controller
 t1 13 controller
 t1 14 controller
 t1 15 controller
 t1 16 controller
 t1 17 controller
 t1 18 controller
 t1 19 controller
 t1 20 controller
 t1 21 controller
 t1 22 controller
 t1 23 controller
 t1 24 controller
 t1 25 controller
 t1 26 controller
 t1 27 controller
 t1 28 controller
```

A typical T1 controller configuration follows:

T1 controller configuration:

```
-----
controller T1 1/0/0:1
 framing esf
 pri-group timeslots 1-24
 controller T1 1/0/0:2
 channel-group 0 timeslots 1-24
 .
 .
```

```

controller T1 1/1/0:28
cas-group 0 timeslots 1-24

```

CT3IP Configuration with Default Values Accepted Example

In the following example, time slots and IP addresses are assigned to channels for the CT3IP in slot 9. (The default framing, cable length, and clock source are accepted for the T3, and the default speed, framing, clock source, and line code are accepted for each T1 channel.)

```

controller t3 9/0/0
t1 16 timeslot 1-24
! Assigns time slots 1 through 24 (the entire T1 bandwidth) to T1 channel 16.
t1 10 timeslot 1-5,20-23
! Assigns time slots 1 through 5 and 20 through 23 (fractional T1 bandwidth)
! to T1 channel 10.
interface serial 9/0/0:16
ip address 10.20.20.1 255.255.255.0
! Assigns IP address 10.20.20.1 to T1 channel 16.
interface serial 9/0/0:10
ip address 10.20.20.3 255.255.255.0
! Assigns IP address 10.20.20.3 to T1 channel 10.
! Other interface configuration commands can be assigned to the T1 channel
! at this time.

```

CT3IP External Ports Configuration Example

In the following example, T1 channel 1 on the CT3IP in slot 9 is broken out as an external port:

```

controller t3 9/0/0
t1 external 1 cablelength 300
! Breaks out T1 channel 1 as an external port so that it can be further channelized on
! the MIP in slot 3.
! Cable length is set to 300 feet.
! The default line coding format (B8ZS) is used for the T1 channel.
controller t1 3/0
linecode b8zs
! The line coding on the MIP is changed to B8ZS to match the line coding on the
! T1 channel.
channel-group 1 timeslots 1
interface serial 3/0:1
ip address 10.20.20.5 255.255.255.0

```

CT3IP Maintenance Data Link Example

The following examples show several of the Maintenance Data Link (MDL) messages for the CT3IP in slot 9:

```

controller t3 9/0/0
mdl string eic Router C
mdl string lic Network A
mdl string fic Bldg 102
mdl string unit 123ABC

```

CT3IP Performance Monitoring Example

In the following example, the performance reports are generated for T1 channel 6 on the CT3IP in slot 9:

```

controller t3 9/0/0
t1 6 fdl ansi

```

BERT Profile Configuration Example

The following example shows a configured BERT profile number 1 that has a 0s test pattern, with a 10⁻² threshold, no error injection, and a duration of 125 minutes:

```
bert profile 1 pattern 0s threshold 10^-2 error-injection none duration 125
```

E2 Clock Rate Configuration Example

The following example shows output when the e2 clock rate is configured using the **e2-clockrate EXEC** command:

```
Router# e2-clockrate
Interface Serial 0 is configured to support clockrates up to E2 (8Mbps)
Interfaces serial 1-3 will not be operational
```

CT3IP BERT Test Pattern Example

The following example shows how to enable a BERT test pattern that consists of a repeating pattern of ones (...111...) and that runs for 30 minutes for T1 channel 8 on CT3IP in slot 9:

```
controller t3 9/0/0
t1 8 bert pattern 1s interval 30
```

CT3IP Remote FDL Loopback Example

The following example shows how to enable a remote payload FDL ANSI bit loopback for T1 channel 6 on CT3IP in slot 3:

```
interface serial 3/0/0:6
loopback remote payload fdl ansi
```

PA-E3 Serial Port Adapter Configuration Example

The following example shows a typical configuration for serial interface 1/0/0 on a PA-E3 serial port adapter in a Cisco 7500 series router:

```
interface serial 1/0/0
ip address 10.1.1.10 255.255.255.0
clock source internal
crc 32
dsu bandwidth 16000
! Reduces the bandwidth by padding the E3 frame.
dsu mode 0
! Enables and improves interoperability with other DSUs.
national bit 1
! Sets bit 12 in the E3 frame to 1.
no scramble
framing g751
no shutdown
```

PA-T3 and PA-2T3 Configuration Example

The following example shows a typical configuration for serial interface 1/0/0 on a PA-T3 serial port adapter in a Cisco 7500 series router:

```
interface serial 1/0/0
 ip address 1.1.1.10 255.255.255.0
 clock source internal
 crc 32
 dsu bandwidth 16000
 ! Reduces the bandwidth by padding the E3 frame.
 dsu mode 0
 ! Enables and improves interoperability with other DSUs.
 no scramble
 framing c-bit
 no shutdown
```

Packet OC-3 Interface Configuration Examples

The examples in this section include a simple configuration and a more complex configuration for two routers back to back.

- [Packet-Over-SONET OC-3 Configuration Example, page 147](#)
- [Packet OC-3 Configuration with Default Values Accepted Example, page 147](#)
- [Two Routers Connected Back-to-Back Example, page 147](#)

Packet-Over-SONET OC-3 Configuration Example

This example shows a POS interface in slot 0, port adapter slot 0, port 0 on a Cisco 7500 series router:

```
interface POS0/0/0
 ip address 10.1.1.4 255.255.255.0
 ip route-cache distributed
 no keepalive
 clock source internal
 pos report rdool
 pos report lais
 pos report lrldi
 pos report pais
 pos report prdi
 pos report sd-ber
 no cdp enable
```

Packet OC-3 Configuration with Default Values Accepted Example

In the following example, the default framing, MTU, and clock source are accepted, and the interface is configured for the IP protocol:

```
interface pos 3/0
 ip address 172.18.2.3 255.0.0.0
```

Two Routers Connected Back-to-Back Example

To connect two routers, attach the cable between the Packet OC-3 port on each. By default, the POS uses loop timing mode. For back-to-back operation, only one of the POSs may be configured to supply its internal clock to the line.

In the following example, two routers are connected back-to-back through their Packet OC-3 interfaces:

First Router

```
interface pos 3/0
 ip address 172.18.2.3 255.0.0.0
 no keepalive
 pos internal-clock
```

Second Router

```
interface pos 3/0
 ip address 172.18.2.4 255.0.0.0
 no keepalive
```

The following example shuts down the entire T1 line physically connected to a Cisco 7500 series routers:

```
controller t1 4/0
 shutdown
```

DPT OC-12c Interface Configuration Examples

This section provides the following configuration examples:

- [DPT Port Adapter Configuration Example, page 148](#)
- [DPT Interface Processor Configuration Example, page 148](#)
- [IPS Options Configuration Example, page 148](#)
- [DPT Topology Configuration Example, page 148](#)

DPT Port Adapter Configuration Example

In the following example, the OC-12c DPT SRP interface is specified, and the IP address and subnet mask are assigned to the interface.

```
interface srp 0/1
 ip address 192.168.2.3 255.255.255.0
```

DPT Interface Processor Configuration Example

In the following example, the OC-12c DPTIP SRP interface is specified, and the IP address and subnet mask is assigned to the interface.

```
interface srp 0/1/0
 ip address 192.168.2.3 255.255.255.0
```

IPS Options Configuration Example

In the following example, the SRP IPS options are configured on a DPT interface:

```
interface srp 2/0
 srp ips request manual-switch a
 srp ips wtr-timer 60
 srp ips timer 90
```

DPT Topology Configuration Example

In the following example, the identity of the nodes on the DPT ring according to their MAC addresses is shown. The following example shows a three-node DPT ring.

```
Router# show srp topology
Topology Map for Interface SRP2/0
Topology pkt. sent every 5 sec. (next pkt. after 4 sec.)
Last received topology pkt. 00:00:00
Nodes on the ring:4
Hops (outer ring)    MAC          IP Address      Wrapped Name
0                    0000.0000.0004 10.2.2.4        No    stingray
1                    0000.0000.0001 10.2.2.1        No    npe300
2                    0000.0000.0005 10.2.2.5        No    gsr
3                    0000.0000.0002 10.2.2.2        No    tuna
```

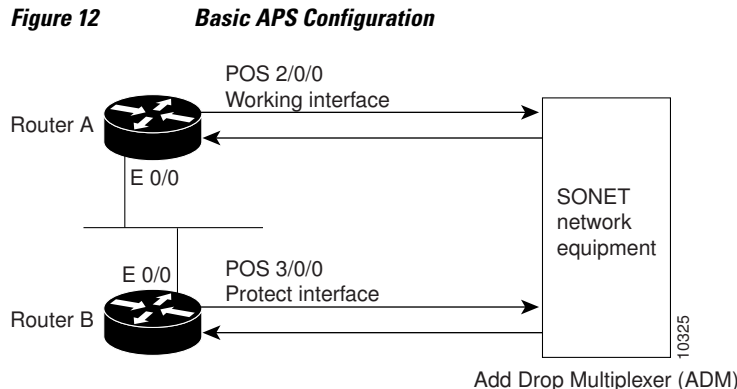
APS Configuration Examples

The following examples show how to configure basic APS on a router and how to configure more than one protect/working interface on a router by using the **aps group** command.

- [Basic APS Configuration Example, page 149](#)
- [Multiple APS Interfaces Configuration Example, page 150](#)

Basic APS Configuration Example

The following example shows the configuration of APS on Router A and Router B (see the figure below). In this example, Router A is configured with the working interface, and Router B is configured with the protect interface. If the working interface on Router A becomes unavailable, the connection will automatically switch over to the protect interface on Router B.



On Router A, which contains the working interface, use the following configuration:

```
interface ethernet 0/0
 ip address 10.7.7.7 255.255.255.0
interface pos 2/0/0
 aps working 1
```

On Router B, which contains the protect interface, use the following configuration:

```
interface ethernet 0/0
 ip address 10.7.7.6 255.255.255.0
interface pos 3/0/0
 aps protect 1 10.7.7.7
```

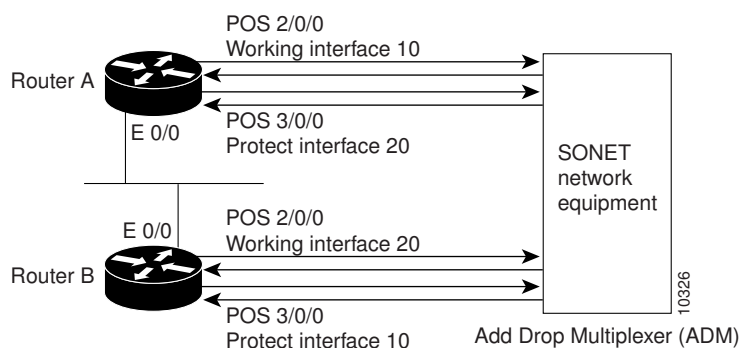
To verify the configuration or to determine if a switchover has occurred, use the **show apscmd** command.

Multiple APS Interfaces Configuration Example

To configure more than one protect/working interface on a router, you must use the **aps group** command. The following example shows the configuration of grouping more than one working/protect interface on a router (see the figure below). In this example, Router A is configured with a working interface and a protect interface, and Router B is configured with a working interface and a protect interface. Consider the following scenarios:

- If the working interface 2/0/0 on Router A becomes unavailable, the connection will switch over to the protect interface 3/0/0 on Router B because they are both in APS group 10.
- If the working interface 2/0/0 on Router B becomes unavailable, the connection will switch over to the protect interface 3/0/0 on Router A because they are both in APS group 20.

Figure 13 Multiple Working and Protect Interfaces Configuration



Note

To avoid the protect interface becoming the active circuit and disabling the working circuit when it is discovered, configure the working interface before configuring the protect interface.

On Router A, which contains the working interface for group 10 and the protect interface for group 20, use the following configuration:

```
interface ethernet 0/0
 ip address 10.7.7.6 255.255.255.0
interface pos 2/0/0
aps group 10
aps working 1
interface pos 3/0/0
aps group 20
aps protect 1 10.7.7.7
```

On Router B, which contains the protect interface for group 10 and the working interface for group 20, use the following configuration:

```
interface ethernet 0/0
 ip address 10.7.7.7 255.255.255.0
interface pos 2/0/0
aps group 20
aps working 1
interface pos 3/0/0
aps group 10
aps protect 1 10.7.7.6
```

To verify the configuration or to determine if a switchover has occurred, use the **show apscmd** command.

CSU DSU Service Module Examples

This section includes three main categories of service module examples:

- [FT1 T1 Examples, page 151](#)
- [2- and 4-Wire 56 64-kbps Service Module Examples, page 153](#)
- [E1-G.703 G.704 Serial Port Adapter Example, page 155](#)

FT1 T1 Examples

FT1/T1 examples are provided for these configurations:

- [T1 Frame Type Example, page 151](#)
- [CSU Line Build-Out Example, page 151](#)
- [T1 Line-Code Type Example, page 151](#)
- [Loop Codes Example, page 151](#)
- [Time Slots Example, page 152](#)
- [Performance Report Example, page 152](#)
- [Loopback Line Enablement Examples, page 152](#)
- [Loopback DTE Examples, page 152](#)
- [Clock Source Example, page 153](#)
- [T1 CSU WIC Configuration Example, page 153](#)

T1 Frame Type Example

The following example enables Superframe as the FT1/T1 frame type:

```
service-module t1 framing sf
```

CSU Line Build-Out Example

The following example shows a line build-out setting of -7.5 dB:

```
service-module t1 lbo -7.5db
```

T1 Line-Code Type Example

The following example specifies AMI as the line-code type:

```
service-module t1 linecode ami
```

Loop Codes Example

The following example displays the configuration of two routers connected back-to-back through an FT1/T1 line and the corresponding feedback messages:

```
no service-module t1 remote-loopback full
service-module t1 remote-loopback payload alternate
loopback remote full
%SERVICE_MODULE-5-LOOPUPFAILED: Unit 0 - Loopup of remote unit failed
service-module t1 remote-loopback payload v54
loopback remote payload
%SERVICE_MODULE-5-LOOPUPFAILED: Unit 0 - Loopup of remote unit failed
```

```
service-module t1 remote-loopback payload alternate
loopback remote payload
%SERVICE_MODULE-5-LOOPUPREMOTE: Unit 0 - Remote unit placed in loopback
```

Time Slots Example

The following example configures a series of time-slot ranges and a speed of 64 kbps:

```
service-module t1 timeslots 1-10,15-20,22 speed 64
```

Performance Report Example

The following is sample output from the **show service-module** serial interface command:

```
Router# show service-module serial 0
Module type is T1/fractional
  Hardware revision is B, Software revision is 1.li,
  Image checksum is 0x21791D6, Protocol revision is 1.1
Receiver has AIS alarm,
Unit is currently in test mode:
  line loopback is in progress
Framing is ESF, Line Code is B8ZS, Current clock source is line,
Fraction has 24 timeslots (64 Kbits/sec each), Net bandwidth is 1536 Kbits/sec.
Last user loopback performed:
  remote loopback
  Failed to loopup remote
Last module self-test (done at startup): Passed
Last clearing of alarm counters 0:05:50
  loss of signal      : 1, last occurred 0:01:50
  loss of frame      : 0,
  AIS alarm          : 1, current duration 0:00:49
  Remote alarm       : 0,
  Module access errors : 0,
Total Data (last 0 15 minute intervals):
  1466 Line Code Violations, 0 Path Code Violations
  0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
  0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
Data in current interval (351 seconds elapsed):
  1466 Line Code Violations, 0 Path Code Violations
  25 Slip Secs, 49 Fr Loss Secs, 40 Line Err Secs, 1 Degraded Mins
  0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 49 Unavail Secs
```

Loopback Line Enablement Examples

The following example shows how to configure a payload loopback:

```
loopback line payload
  Loopback in progress
no loopback line
```

The following example shows the output when you loop a packet in switched mode without an active connection:

```
service-module 56k network-type switched
loopback line payload
  Need active connection for this type of loopback
  % Service module configuration command failed: WRONG FORMAT.
```

Loopback DTE Examples

The following example loops a packet from a module to the serial interface:

```
loopback dte
  Loopback in progress
```

```
ping 10.0.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 10.0.0.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/12/28 ms
```

Clock Source Example

The following example shows a router using internal clocking while transmitting frames at 38.4 kbps:

```
service-module 56k clock source internal
service-module 56k clock rate 38.4
```

TI CSU WIC Configuration Example

The following example shows how to set the **fdl** parameter to **att** while in interface configuration mode:

```
interface Serial0/0
no ip address
no ip route-cache
no ip mroute-cache
no keepalive
shutdown
no fair-queue
service-module t1 clock source internal
service-module t1 fdl att
no cdp enable
```

2- and 4-Wire 56 64-kbps Service Module Examples

This section provides the following examples for 2- and 4-wire, 56/64-kbps service modules:

- [Network Line Speed Examples, page 153](#)
- [Scrambled Data Coding Example, page 154](#)
- [Switched Dial-Up Mode Example, page 154](#)
- [Performance Report Example, page 154](#)
- [Remote Loopback Request Example, page 154](#)
- [Service Provider Example, page 155](#)

Network Line Speed Examples

The following example displays the configuration of two routers connected in back-to-back DDS mode. However, the configuration fails because the **auto** rate is used, which is not valid in back-to-back mode.

```
Router1# service-module 56k clock source internal
Router1# service-module 56k clock rate 38.4
Router2# service-module 56k clock rate auto
% WARNING - auto rate will not work in back-to-back DDS.
a1# ping 10.1.1.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
Router2# service-module 56k clock rate 38.4
Router1# ping 10.1.1.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 52/54/56 ms
```

When transferring from DDS mode to switched mode, you must set the correct clock rate, as shown in the following example:

```
service-module 56k network-type dds
service-module 56k clock rate 38.4
service-module 56k network-type switched
% Have to use 56k or auto clock rate for switched mode
% Service module configuration command failed: WRONG FORMAT.
service-module 56k clock rate auto
% WARNING - auto rate will not work in back-to-back DDS.
service-module 56k network-type switched
```

Scrambled Data Coding Example

The following example scrambles bit codes in 64-kbps DDS mode:

```
service-module 56k clock rate 56
service-module 56k data-coding scrambled
Can configure scrambler only in 64k speed DDS mode
% Service module configuration command failed: WRONG FORMAT.
service-module 56k clock rate 64
service-module 56k data-coding scrambled
```

Switched Dial-Up Mode Example

The following example displays transmission in switched dial-up mode:

```
service-module 56k clock rate 19.2
service-module 56k network-type switched
% Have to use 56k or auto clock rate for switched mode
% Service module configuration command failed: WRONG FORMAT.
service-module 56k clock rate auto
service-module 56k network-type switched
dialer in-band
dialer string 2576666
dialer-group 1
```

Performance Report Example

The following is sample output from the **show service-module serial** command:

```
Router# show service-module serial 1
Module type is 4-wire Switched 56
  Hardware revision is B, Software revision is X.07,
  Image checksum is 0x45354643, Protocol revision is 1.0
Connection state: active,
Receiver has loss of signal, loss of sealing current,
Unit is currently in test mode:
  line loopback is in progress
Current line rate is 56 Kbits/sec
Last user loopback performed:
  dte loopback
  duration 00:00:58
Last module self-test (done at startup): Passed
Last clearing of alarm counters 0:13:54
  oos/oof          :    3, last occurred 0:00:24
  loss of signal   :    3, current duration 0:00:24
  loss of sealing curren:    2, current duration 0:04:39
  loss of frame    :    0,
  rate adaption attempts:    0,
```

Remote Loopback Request Example

The following example enables you to transmit and receive remote loopbacks:

```
service-module 56k remote-loopback
```

Service Provider Example

The following example selects AT&T as the service provider:

```
service-module 56k network-type switched
service-module 56k switched-carrier att
```

E1-G.703 G.704 Serial Port Adapter Example

The following example shows a configuration for serial interface 9/1/3 on a E1-G.703/G.704 serial port adapter in a Cisco 7500 series router. In this example, the interface is configured for framed (G.704) operation, and time slot 16 is used for data.

```
interface serial 9/1/3
 ip address 10.1.1.10 255.255.255.0
 no keepalive
 no fair-queue
 timeslot 1-31
 crc4
 ts16
```

Low-Speed Serial Interface Examples

The section includes the following configuration examples for low-speed serial interfaces:

- [Synchronous or Asynchronous Mode Examples, page 155](#)
- [Controlled-Carrier and Constant-Carrier Mode Examples, page 156](#)
- [Half-Duplex Timers Example, page 156](#)
- [Cisco 4000 Series Router with 2T16S Serial Network Processor Examples, page 156](#)

Synchronous or Asynchronous Mode Examples

The following example shows how to change a low-speed serial interface from synchronous to asynchronous mode:

```
interface serial 2
 physical-layer async
```

The following examples show how to change a low-speed serial interface from asynchronous mode back to its default synchronous mode:

```
interface serial 2
 physical-layer sync
```

or

```
interface serial 2
 no physical-layer
```

The following example shows some typical asynchronous interface configuration commands:

```
interface serial 2
 physical-layer async
 ip address 10.0.0.2 255.0.0.0
```

```

async default ip address 10.0.0.1
async mode dedicated
async default routing

```

The following example shows some typical synchronous serial interface configuration commands available when the interface is in synchronous mode:

```

interface serial 2
physical-layer sync
ip address 10.0.0.2 255.0.0.0
no keepalive
ignore-dcd
nrzi-encoding
no shutdown

```

Controlled-Carrier and Constant-Carrier Mode Examples

The following example shows how to change to controlled-carrier mode from the default of constant-carrier operation:

```

interface serial 2
half-duplex controlled-carrier

```

The following example shows how to change to constant-carrier mode from controlled-carrier mode:

```

interface serial 2
no half-duplex controlled-carrier

```

Half-Duplex Timers Example

The following example shows how to set the cts-delay timer to 1234 ms and the transmit-delay timer to 50 ms:

```

interface serial 2
half-duplex timer cts-delay 1234
half-duplex timer transmit-delay 50

```

Cisco 4000 Series Router with 2T16S Serial Network Processor Examples

The 2T16S network processor module provides high-density serial interfaces for the Cisco 4000 series routers. This module has two high-speed interfaces that support full-duplex T1 and E1 rates (up to 2 MB per second) and 16 low-speed interfaces. The 16 lower-speed ports can be individually configured as either as synchronous ports at speeds up to 128 kbps or as asynchronous ports at speeds up to 115 kbps.

For the low-speed interfaces, both synchronous and asynchronous serial protocols are supported. For the high-speed interfaces, only the synchronous protocols are supported. Synchronous protocols include IBM's Bisync, SDLC, and HDLC. Asynchronous protocols include PPP, SLIP, and ARAP for dial-up connections using external modems.

The following example shows a Cisco 4500 router equipped with two 2T16S serial network processor modules and two conventional Ethernet ports. The router is configured for WAN aggregation using X.25, Frame Relay, PPP, and HDLC encapsulation. Serial interfaces 0, 1, 18, and 19 are the synchronous high-speed interfaces. Serial interfaces 2 through 17 and 20 through 35 are the synchronous/asynchronous low-speed interfaces.

```

version 11.2
!
hostname c4X00
!
username brad password 7 13171F1D0A080139

```

```
username jim password 7 104D000A0618
!
```

Ethernet interfaces and their subinterfaces are configured for LAN access.

```
interface Ethernet0
 ip address 10.1.1.1 255.255.255.0
 media-type 10BaseT
!
interface Ethernet1
 ip address 10.1.2.1 255.255.255.0
 media-type 10BaseT
!
```

Serial interfaces 0 and 1 are the high-speed serial interfaces on the first 2T16S module. In this example, subinterfaces are also configured for remote offices connected in to serial interface 0:

```
interface Serial0
 description Frame Relay configuration sample
 no ip address
 encapsulation frame-relay
!
interface Serial0.1 point-to-point
 description PVC to first office
 ip address 10.1.3.1 255.255.255.0
 frame-relay interface-dlci 16
!
interface Serial0.2 point-to-point
 description PVC to second office
 ip address 10.1.4.1 255.255.255.0
 frame-relay interface-dlci 17
!
interface Serial1
 description X25 configuration sample
 ip address 10.1.5.1 255.255.255.0
 no ip mroute-cache
 encapsulation x25
 x25 address 6120184321
 x25 htc 25
 x25 map ip 10.1.5.2 6121230073
```

Serial interfaces 2 to 17 are the low-speed interfaces on the 2T16S network processor module. In this example, remote routers are connected to various configurations.

```
interface Serial2
 description DDR connection router dial out to remote sites only
 ip address 10.1.6.1 255.255.255.0
 dialer in-band
 dialer wait-for-carrier-time 60
 dialer string 0118527351234
 pulse-time 1
 dialer-group 1
!
interface Serial3
 description DDR interface to answer calls from remote office
 ip address 10.1.7.1 255.255.255.0
 dialer in-band
!
interface Serial4
 description configuration for PPP interface
 ip address 10.1.8.1 255.255.255.0
 encapsulation ppp
!
interface Serial5
 description Frame Relay configuration sample
 no ip address
 encapsulation frame-relay
!
interface Serial5.1 point-to-point
 description PVC to first office
 ip address 10.1.9.1 255.255.255.0
```

```

    frame-relay interface-dlci 16
  !
interface Serial5.2 point-to-point
  description PVC to second office
  ip address 10.1.10.1 255.255.255.0
  frame-relay interface-dlci 17
  !
interface Serial6
  description Configuration for PPP interface
  ip address 10.1.11.1 255.255.255.0
  encapsulation ppp
  !
interface Serial7
  no ip address
  shutdown
  !
interface Serial8
  ip address 10.1.12.1 255.255.255.0
  encapsulation ppp
  async default routing
  async mode dedicated
  !
interface Serial9
  physical-layer async
  ip address 10.1.13.1 255.255.255.0
  encapsulation ppp
  async default routing
  async mode dedicated
  !
interface Serial10
  physical-layer async
  no ip address
  !
interface Serial11
  no ip address
  shutdown
  !
interface Serial12
  physical-layer async
  no ip address
  shutdown
  !
interface Serial13
  no ip address
  shutdown
  !
interface Serial14
  no ip address
  shutdown
  !
interface Serial15
  no ip address
  shutdown
  !
interface Serial16
  no ip address
  shutdown
  !
interface Serial17
  no ip address
  shutdown

```

Serial interface serial 18 is the first high-speed serial interface of the second 2T16S module. Remote sites on different subnets are dialing in to this interface with point-to-point and multipoint connections.

```

interface Serial18
  description Frame Relay sample
  no ip address
  encapsulation frame-relay
  !
interface Serial18.1 point-to-point
  description Frame Relay subinterface
  ip address 10.1.14.1 255.255.255.0

```

```

    frame-relay interface-dlci 16
  !
interface Serial18.2 point-to-point
  description Frame Relay subinterface
  ip address 10.1.15.1 255.255.255.0
  frame-relay interface-dlci 17
  !
interface Serial18.3 point-to-point
  description Frame Relay subinterface
  ip address 10.1.16.1 255.255.255.0
  frame-relay interface-dlci 18
  !
interface Serial18.5 multipoint
  ip address 10.1.17.1 255.255.255.0
  frame-relay map ip 10.1.17.2 100 IETF

```

This second high-speed serial interface is configured to connect a X.25 link. Serial interfaces 20 through 35 are the low-speed interfaces. However, some of the interfaces are not displayed in this example.

```

interface Serial19
  description X25 sample configuration
  ip address 10.1.18.1 255.255.255.0
  no ip mroute-cache
  encapsulation x25
  x25 address 6120000044
  x25 htc 25
  x25 map ip 10.1.18.2 6120170073
  !
interface Serial20
  ip address 10.1.19.1 255.255.255.0
  !
interface Serial21
  physical-layer async
  ip unnumbered e0
  encapsulation ppp
  async mode dedicated
  async dynamic routing
  ipx network 45
  ipx watchdog-spoof
  dialer in-band
  dialer-group 1
  ppp authentication chap
  !
interface Serial22
  no ip address
  shutdown
  !
interface Serial23
  no ip address
  shutdown
  !
interface Serial24
  no ip address
  shutdown
  !
! Serial interfaces 23 through 35 would appear here.
.
.
.
router eigrp 10
  network 10.0.0.0
  !
dialer-list 1 protocol ip permit
  !
line con 0
  exec-timeout 15 0
  password david
  login

```

The following basic line example configures some of the low-speed serial interfaces for the module:

```
line 8 10
```

```

modem InOut
transport input all
rxspeed 64000
txspeed 64000
flowcontrol hardware
line 12
transport input all
rxspeed 64000
txspeed 64000
flowcontrol hardware
modem chat-script generic
line 21
transport input all
rxspeed 64000
txspeed 64000
flowcontrol hardware
!
end

```

Additional References

The following sections provide references related to the Configuring Serial Interfaces feature.

Related Documents

Related Topic	Document Title
encapsulation, invert data, and other interface commands	<i>Cisco IOS Interface and Hardware Component Command Reference</i>
Configuring Media-Independent PPP and Multilink PPP	<i>Cisco IOS Dial Technologies Configuration Guide</i>
Configuring Serial Tunnel and Block Serial Tunnel	<i>Cisco IOS Bridging and IBM Networking Configuration Guide</i>
Configuring L2TPv3	<i>Cisco IOS Multiprotocol Label Switching Configuration Guide</i>

Standards

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

MIBs

MIB	MIBs Link
<ul style="list-style-type: none"> • T1 MIB • T3 MIB • E3 MIB • DS3 MIB • Standard MIB II • DSU/CSU MIB 	<p>To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:</p> <p>http://www.cisco.com/go/mibs</p>

RFCs

RFC	Title
RFC 1889	<i>RTP--A Transport Protocol for Real-Time Applications</i>
RFC 1406	<i>Definitions of Managed Objects for DS1 and E1 Interface Types</i>
RFC 1407	<i>DS3 MIB Variables</i>
RFC 1619	<i>PPP over SONET/SDH</i>
RFC 1662	<i>PPP in HDLC-like Framing</i>

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/techsupport</p>

Feature Information for Configuring Serial Interfaces

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 10 *Feature Information for Configuring Serial Interfaces*

Feature Name	Releases	Feature Information
Configuring Serial Interfaces	12.1 12.2S 12.4T 15.0(1)M	This module describes the serial interfaces on routers supporting Cisco IOS software. No new commands were introduced or modified.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



Circuit Emulation over IP

Circuit Emulation over IP (CEoIP) provides a virtual circuit through an IP network--similar to a leased line--to integrate solutions that require a time-sensitive, bit-transparent transport into IP networks. Data, with proprietary framing or without, arrives at its destination unchanged; the transport is transparent to the destination.

- [Finding Feature Information, page 163](#)
- [Prerequisites for Circuit Emulation over IP, page 163](#)
- [Restrictions for Circuit Emulation over IP, page 163](#)
- [Information About Circuit Emulation over IP, page 164](#)
- [How to Configure Circuit Emulation over IP, page 166](#)
- [Configuration Examples for CEoIP, page 179](#)
- [Additional References, page 180](#)
- [Command Reference, page 181](#)
- [Feature Information for Circuit Emulation over IP, page 182](#)

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.

Prerequisites for Circuit Emulation over IP

- The CEoIP feature requires a CEoIP network module (NM) on each end of the connection, either the NM-CEM-4TE1 NM or the NM-CEM-4SER NM. You do not need to use the same type of CEoIP NM on both ends of the connection.
- The CEoIP feature requires 300 KB of flash memory and 1 MB of DRAM in addition to your Cisco IOS software requirements.

Restrictions for Circuit Emulation over IP

- NM-CEM-4TE1 supports only B8ZS (T1) and HDB3 (E1) line codes.
- E1 lines do not support 56 kbps connections.
- CEoIP software cannot run payload compression for more than 3.088 Mbps) per network module.
- If you configure four T1, E1, or serial cables (over 1.544 M) at the same time in Cisco 2600XM series routers, you cannot turn on the data-protection and payload compression features. Also, in framed mode (channelized), you can use up to 60 channels without the data protection and payload compression features on Cisco 2600XM series routers. However, you can turn on the data protection and payload compression feature in one T1/E1.
- There is a limitation on the data protection and payload compression features on Cisco 3660 routers. If you configure four T1, E1, or serial cables on Cisco 3660 routers, you can turn on data protection for up to two T1/E1s. In framed mode, you can use 88 channels.

Information About Circuit Emulation over IP

- [Circuit Emulation over IP, page 164](#)
- [Benefits of CEM over IP, page 165](#)
- [Adaptive Clocking for CEoIP, page 165](#)
- [Clock Switchover, page 165](#)
- [Payload Compression for CEoIP, page 165](#)
- [Data Protection \(Sample Repetition\), page 165](#)
- [Dejitter, page 165](#)
- [Idle Pattern, page 166](#)
- [Payload Size, page 166](#)
- [Signaling for CEoIP, page 166](#)
- [Control Lead Configurations, page 166](#)

Circuit Emulation over IP

Circuit emulation is an end-to-end service that allows Layer 1 data to be transported transparently through an IP network. Applications that require circuit emulation need the network to provide a constant rate bit stream.

Configuration Circuit Emulation (CEM) can be configured in unidirectional mode using the emulation-mode option. Once configured, traffic will flow only in that direction through the CEM channel. When one direction of CEM traffic is detected on that channel, the CEM channel is considered to be active and a new status of the CEM channel is created to reflect the uni-directional channel.

CEoIP may use adaptive clocking as a means of synchronizing the clock frequencies at the two endpoints. Channel associated signaling (CAS) transport is provided as an optional feature to allow channelized voice applications. Payload compression is provided as an optional feature to improve bandwidth efficiency and data protection is provided to reduce the probability of data loss.

CEoIP software supports the following network modules:

- The NM-CEM-4SER, a network module with four serial ports. To configure CEoIP software for the NM-CEM-4SER, you must configure the options of the ports. Options include dejitter buffer, payload compression, and payload size.
- The NM-CEM-4TE1, a network module with four ports that you can configure as T1 or E1 (where all four ports support the same interface type). To configure CEoIP software for the NM-CEM-TE1, you must define the card type and then configure the options of the port.

Benefits of CEM over IP

CEoIP provides a simple migration path to IP-only networks. Examples of solutions that CEoIP integrates with IP include the following:

- Legacy data services
- Legacy video applications
- Satellite data streams
- Radar data streams
- Telemetry for automated industrial environments (for example, power distribution)
- Crypto tunneling for multilevel security

Adaptive Clocking for CEoIP

The adaptive clocking option of CEoIP allows the egress clock to vary by expanding or contracting the clock period from the nominal clock. After you have implemented the clocking feature, the adaptive clocking circuits continuously adjust the selected clock based on the data buffer level. You can implement adaptive clocking on each port independently.

Clock Switchover

The clock switchover option allows you to switch the clock source over to the internal clock. The switchover ensures continuity of the CEM channel when disruption in receiving the clock from the customer premises equipment (CPE) occurs. To specify the input lead state change that triggers the clock switching over from line to internal or from internal to line, use this option in Data Circuit Terminating Equipment (DCE) split mode.

Payload Compression for CEoIP

The payload compression option minimizes the amount of bandwidth that traffic consumes. It compresses the transmission of any repetitive data pattern (for example, idle code, HDLC flags, and so on) to increase the efficiency of the solution across the network.

With CEoIP software, you can adjust the size (in bytes) of the payload for the IP packet to configure efficiency as opposed to packetization. Larger payloads provide more efficiency but increase the delay. With smaller packets the overhead of the header increases. Payload compression is disabled by default.

Data Protection (Sample Repetition)

The data protection option, also known as sample repetition, reduces the probability of errors due to packet loss by sending each sample twice, in two different IP packets. Data protection consumes more bandwidth than standard transmission, but you can minimize the amount of traffic with payload compression. This feature is disabled by default.

Dejitter

The dejitter buffer size determines the ability of the emulated circuit to tolerate network jitter. The dejitter buffer in CEoIP software is configurable up to 500 milliseconds; the maximum amount of network jitter that CEoIP can tolerate is ± 250 milliseconds.

Idle Pattern

The idle pattern option specifies the idle pattern to transmit when the circuit goes down. You can specify a maximum of 64 bits with two 32-bit patterns for the NM-CEM-4SER and 8-bit patterns for the NM-CEM-4TE1.

Payload Size

Payload size is the number of bytes put into each IP packet. This parameter impacts packetization delay and efficiency. Configure a high payload size to increase packetization delay and efficiency. A smaller payload size reduces packetization delay and efficiency.

Signaling for CEoIP

CEoIP software supports the transport of channel associated signaling (CAS) bits in channelized T1/E1 mode. This option extracts incremental signaling information and sends that information in separate packets.

Control Lead Configurations

CEoIP software supports the monitoring and transport of serial interface control leads.

How to Configure Circuit Emulation over IP

- [Configuring the NM-CEM-4TE1 Card Type, page 166](#)
- [Configuring the T1 E1 Line, page 167](#)
- [Creating CEM Channels on the T1 E1 Line, page 170](#)
- [Configuring the Connection Using the xconnect Command, page 171](#)
- [Configuring the CEM Channel, page 173](#)

Configuring the NM-CEM-4TE1 Card Type

Perform this task to configure the card type for an NM-CEM-4TE1.

This task does not apply to the NM-CEM-4SER.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **card type {t1 | e1} slot**

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 <code>enable</code></p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
<p>Step 2 <code>configure terminal</code></p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
<p>Step 3 <code>card type {t1 e1} slot</code></p> <p>Example:</p> <pre>Router(config)# card type t1 1</pre>	<p>Configures the card type by specifying the transmission mode for the ports on the network module.</p> <ul style="list-style-type: none"> All four ports on the CEoIP T1/E1 network module must operate in the same mode. Use the t1 or e1 keyword to specify the transmission mode for all four ports. <p>Note This command is only entered once and changes do not take effect unless the reload command is used, or the router is rebooted.</p>

- [What to Do Next, page 167](#)

What to Do Next

Go to the [Configuring the T1 E1 Line, page 167](#) to continue configuring CEoIP on an NM-CEM-4TE1.

Configuring the T1 E1 Line

Perform this task to configure the T1 or E1 line, starting in global configuration mode.

This task does not apply to the NM-CEM-4SER.

SUMMARY STEPS

1. **controller** {**t1** | **e1**} *slot / port*
2. Do one of the following:
 - **framing** {**esf** | **sf** | **unframed**}
 -
 -
 - **framing** {**crc4** | **no-crc4** | **unframed**}
3. **clock source** {**internal** | **line**| **adaptive** *channel-number*}
4. **cablelength** {**long** | **short**} {*attenuation*| *length*}
5. **crc-threshold** *value*
6. **description** *text*
7. **loopback** {**local** {**line** | **payload**}}| **network**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>controller {t1 e1} <i>slot / port</i></p> <p>Example:</p> <pre>Router(config)# controller t1 1/0</pre>	<p>Enters controller configuration mode.</p> <ul style="list-style-type: none"> • Use the <i>slotand port</i> arguments to specify the slot number and port number to be configured.
Step 2	<p>Do one of the following:</p> <ul style="list-style-type: none"> • framing {esf sf unframed} • • • framing {crc4 no-crc4 unframed} <p>Example:</p> <pre>Router(config-controller)# framing esf</pre> <p>Example:</p> <pre>Router(config-controller)# framing crc4</pre>	<p>(Optional) Configures the framing format for a T1 or E1 port to synchronize the port and the attached device.</p> <p>T1 port framing options:</p> <ul style="list-style-type: none"> • Use the esf keyword to specify Extended Superframe as the T1 framing type. • Use the sf keyword to specify the Superframe (also commonly called D4 framing) as the T1 framing type. This is the default. <p>E1 port framing options:</p> <ul style="list-style-type: none"> • Use the crc4 keyword to specify the G.704 standard with optional CRC4 mechanism defined in timeslot zero (0) enabled as the E1 framing type. This is the default. • Use the no-crc4 keyword to specify the G.704 standard with optional CRC4 mechanism defined in timeslot zero (0) disabled as the E1 framing type. <p>T1 or E1 port framing options:</p> <ul style="list-style-type: none"> • Use the unframed keyword to specify the unchannelized mode of framing. <p>Note If you do not configure framing, the framing on the customer premises equipment (CPE) devices on each end of the connection must match.</p>

Command or Action	Purpose
<p>Step 3 <code>clock source {internal line} adaptive channel-number</code></p> <p>Example:</p> <pre>Router(config-controller)# clock source adaptive 6</pre>	<p>Configures the clock source for a T1 or E1 port.</p> <ul style="list-style-type: none"> Use the internal keyword to specify that the port transmit clock is derived from the time-division multiplexing (TDM) bus backplane clock, if one exists in the router, or the on-board oscillator on the network module. Use the line keyword to specify that the port transmit clock is derived from the receive clock on the same port. Use the adaptive keyword to specify that the port transmit clock is locally synthesized based on the average data content of the dejitter buffer of one of the channels on this port. If the adaptive keyword is selected, use the <i>channel-number</i> argument to specify the channel whose dejitter buffer is to be used to synthesize the transmit clock of the port.
<p>Step 4 <code>cablelength {long short} {attenuation} length</code></p> <p>Example:</p> <pre>Router(config-controller)# cablelength long -15db</pre>	<p>(Optional) Specifies the line build-out characteristics of the internal CSU on a T1 port.</p> <ul style="list-style-type: none"> Use the long keyword to specify that the signal characteristics are set for a long cable length. Use the short keyword to specify that the signal characteristics are set for a short cable length. If the long keyword is selected, use the <i>attenuation</i> argument to specify the T1 signal attenuation. If the short keyword is selected, use the <i>length</i> argument to specify the T1 cable length. <p>Note This command does not apply to an E1 port.</p>
<p>Step 5 <code>crc-threshold value</code></p> <p>Example:</p> <pre>Router(config-controller)# crc-threshold 512</pre>	<p>(Optional) Configures the number of cyclical redundancy check (CRC) errors in one second that results in the second being declared as a Severely Errored Second (SES).</p> <ul style="list-style-type: none"> Use the <i>value</i> argument to specify the number of CRC errors. Default is 320. <p>Note This command does not apply to an E1 port.</p>
<p>Step 6 <code>description text</code></p> <p>Example:</p> <pre>Router(config-controller)# description T1 line to 3rd floor PBX</pre>	<p>(Optional) Specifies a text description of the port.</p>

Command or Action	Purpose
<p>Step 7 <code>loopback {local {line payload} network}</code></p> <p>Example:</p> <pre>Router(config-controller)# loopback network</pre>	<p>(Optional) Creates a loopback from a T1 or E1 port.</p> <ul style="list-style-type: none"> Use the local keyword to create a loopback where the information from a locally-attached CPE is transmitted back to the locally-attached CPE. Use the network keyword to create a loopback where the data received over the network from a remotely-attached CPE is transmitted back to the remotely-attached CPE. If the local keyword is selected, use the line keyword to create a full physical layer loopback of all bits, including data and framing. If the local keyword is selected, use the payload keyword to create a loopback of the data in the individual timeslots only. In this mode, framing bits are terminated on entry and regenerated on exit instead of being looped back. This mode is not available if the port is configured for framing unframed.

- [What to Do Next, page 170](#)

What to Do Next

Go to the [Creating CEM Channels on the T1 E1 Line, page 170](#) to continue configuring CEoIP on an NM-CEM-4TE1

Creating CEM Channels on the T1 E1 Line

Perform this task to create CEM channels on the T1 or E1 line, starting in controller configuration mode.

This task does not apply to the NM-CEM-4SER.

SUMMARY STEPS

1. `cem-group group-number {unframed | timeslots timeslot [speed {56 | 64}]}`
2. `exit`

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 <code>cem-group</code> <i>group-number</i> {<code>unframed</code> <code>timeslots</code> <i>timeslot</i> [<code>speed</code> {<code>56</code> <code>64</code>}]}</p> <p>Example:</p> <pre>Router(config-controller)# cem- group 6 timeslots 1-4,9,10 speed 64</pre>	<p>Creates a circuit emulation channel from one or more timeslots of a T1 or E1 line of an NM-CEM-4TE1.</p> <ul style="list-style-type: none"> • The <i>group-number</i> keyword identifies the channel number to be used for this channel. For T1 ports, the range is 0-23. For E1 ports, the range is 0-30. • Use the unframed keyword to specify that a single CEM channel is being created including all timeslots and the framing structure of the line. • Use the timeslots keyword and the <i>timeslot-range</i> argument to specify the timeslots to be included in the CEM channel. The list of timeslots may include commas and hyphens with no spaces between the numbers, commas, and hyphens. • Use the speed keyword to specify the speed of the channels by specifying the number of bits of each timeslot to be used. This keyword applies only to T1 channels.
<p>Step 2 <code>exit</code></p> <p>Example:</p> <pre>Router(config-controller)# exit</pre> <p>Example:</p> <pre>Router(config)#</pre>	<p>Exits controller configuration mode and returns to global configuration mode.</p>

- [What to Do Next, page 171](#)

What to Do Next

Go to the [Configuring the Connection Using the xconnect Command, page 171](#) to continue configuring CEoIP on an NM-CEM-4TE1

Configuring the Connection Using the xconnect Command

Perform this task to create a connection using the **xconnect** command, starting in global configuration mode. This task applies to configuring CEoIP on both the NM-CEM-4TE1 and the NM-CEM-4SER.



Note

To properly configure the CEoIP feature, two CEoIP network modules must use the same UDP port number to communicate.

SUMMARY STEPS

1. **cem slot / port / cem-group**
2. **xconnect remote-ip-address virtual-connect-ID encapsulation encapsulation-type**
3. **local ip address local-ip-address**
4. **local udp port udp-port**
5. **remote udp port udp-port**
6. **exit**

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 cem slot / port / cem-group</p> <p>Example:</p> <pre>Router(config)# cem 3/1/0</pre>	<p>Enters CEM configuration mode to configure CEM channels.</p> <ul style="list-style-type: none"> • Use the <i>slot</i> argument to specify the slot number in which the network module is installed. • Use the <i>port</i> argument to specify the port number of the CEM channel to be configured. • Use the <i>channel</i> argument to specify the CEM channel number to be configured. For a serial channel enter zero. For a T1 or E1 channel enter the channel number defined in the cem-group command (see the Creating CEM Channels on the T1 E1 Line, page 170).
<p>Step 2 xconnect remote-ip-address virtual-connect-ID encapsulation encapsulation-type</p> <p>Example:</p> <pre>Router(config-cem)# xconnect 10.2.0.1 0 encapsulation udp</pre>	<p>Creates one end of a connection between two CEM network modules and enters xconnect configuration mode.</p> <ul style="list-style-type: none"> • Use the <i>remote-ip-address</i> argument to specify the IP address of an interface--regular or loopback--on the destination router. • Set the <i>virtual-connect-ID</i> argument to be zero. <p>Note Currently the only supported encapsulation type is UDP.</p>
<p>Step 3 local ip address local-ip-address</p> <p>Example:</p> <pre>Router(config-cem-xconnect)# local ip-address 10.2.0.2</pre>	<p>Configures the IP address of an interface--regular or loopback--on the source router.</p> <p>Note The local IP address must be the same as the remote IP address (at the other side) configured in the xconnect command.</p>
<p>Step 4 local udp port udp-port</p> <p>Example:</p> <pre>Router(config-cem-xconnect)# local udp port 15901</pre>	<p>Specifies the User Datagram Protocol (UDP) port number of the local CEM channel.</p> <p>Note The local UDP port of a CEM channel must be the same as the remote UDP port of the CEM channel at the other end of the connection.</p>

Command or Action	Purpose
<p>Step 5 <code>remote udp port <i>udp-port</i></code></p> <p>Example:</p> <pre>Router(config-cem-xconnect)# remote udp port 15902</pre>	<p>Specifies the UDP port number of the remote CEM channel.</p> <p>Note The remote UDP port of a CEM channel must be the same as the local UDP port of the CEM channel at the other end of the connection.</p>
<p>Step 6 <code>exit</code></p> <p>Example:</p> <pre>Router(config-cem-xconnect)# exit</pre> <p>Example:</p> <pre>Router(config-cem)#</pre>	<p>Exits xconnect configuration mode and returns to CEM configuration mode.</p> <ul style="list-style-type: none"> Repeat this command if you wish to exit CEM configuration mode return to global configuration mode.

- [What to Do Next, page 173](#)

What to Do Next

This task must be repeated on the other CEM network module and each end of the CEM connection must be configured identically to allow traffic to pass between the network modules. When both network modules have been configured, continue to the [Configuring the CEM Channel, page 173](#).

Configuring the CEM Channel

Perform this task to configure the CEM T1/E1 or serial channel, starting in CEM configuration mode.

SUMMARY STEPS

1. **clock rate** *rate*
2. **clock mode** { **normal** | **split** }
3. **clock source** { **internal** | **loop** | **adaptive** }
4. **payload-size** *size*
5. **de jitter-buffer** *size*
6. **control-lead sampling-rate** *rate*
7. **control-lead state** { **active** | **fail** } *output-lead*{ **on** | **off** | **follow** } [{ **local** | **remote** } *input-lead*]
8. **data-strobe** *input-lead* { **on** | **off** }
9. Cisco NM-CEM-4SER:
10. **signaling**
11. **payload-compression**
12. **data-protection**
13. **ip dscp** *dscp*
14. **ip tos** *tos*
15. **ip precedence** *precedence*
16. **loopback** { **local** | **network** }
17. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	clock rate <i>rate</i> Example: Router(config-cem)# clock rate 38400	(Optional) For serial channels only. Specifies the nominal bit rate of a serial CEM channel. <ul style="list-style-type: none"> • Use the <i>rate</i> argument to specify the data rate of the channel in bps. Default is 64000.
Step 2	clock mode { normal split } Example: Router(config-cem)# clock mode split	(Optional) For serial channels only. Specifies the clock mode of a serial CEM channel. <ul style="list-style-type: none"> • Use the normal keyword to specify that the Data Circuit-terminating Equipment (DCE) provides both the Receive Clock (RxC) and the Transmit clock (TxC) to the attached Data Terminal Equipment (DTE). • Use the split keyword to specify that the DCE provides the Receive Clock (RxC) to the attached DTE, and the DTE provides the external Transmit Clock (XTC or TT) to the DCE. <p>Note Depending on the serial cable attached to the port, the port is automatically configured as either a DCE or DTE.</p>

Command or Action	Purpose
<p>Step 3 <code>clock source {internal loop adaptive}</code></p> <p>Example:</p> <pre>Router(config-cem)# clock source loop</pre>	<p>(Optional) Configures the clock source for a serial CEM channel.</p> <ul style="list-style-type: none"> This step applies only to configuring serial channels. For information about configuring the clock source for T1 or E1 ports, see the Configuring the T1 E1 Line, page 167. Use the internal keyword to specify that the clock(s) provided by the network module to the CPE is derived from the TDM bus backplane clock, if one exists in the router, or the on-board oscillator on the network module. Use the loop keyword to specify that the clock provided by the network module to the CPE is derived from the the clock receive from the CPE on the same port. Use the adaptive keyword to specify that the clock(s) provided by the network module to the CPE is locally synthesized based on the average data content of the local dejitter buffer. <p>Note The loop keyword is valid only when the clock mode split command is configured.</p>
<p>Step 4 <code>payload-size size</code></p> <p>Example:</p> <pre>Router(config-cem)# payload-size 512</pre>	<p>(Optional) Specifies the number of bytes encapsulated into a single IP packet.</p> <ul style="list-style-type: none"> Use the <i>size</i> argument to specify the number of bytes included in the payload of each packet. Default is 32 bytes for a serial CEM channel. For more information about T1 and E1 default values, see the payload-size command in the "Command Reference" section.
<p>Step 5 <code>dejitter-buffer size</code></p> <p>Example:</p> <pre>Router(config-cem)# dejitter-buffer 80</pre>	<p>(Optional) Specifies the size of the dejitter buffer used to compensate for the network filter.</p> <ul style="list-style-type: none"> Use the <i>size</i> argument to specify the size of the buffer in milliseconds. Default is 60.
<p>Step 6 <code>control-lead sampling-rate rate</code></p> <p>Example:</p> <pre>Router(config-cem)# control-lead sampling-rate 10</pre>	<p>(Optional) Specifies the sampling rate of input control leads on a serial CEM channel.</p> <ul style="list-style-type: none"> This command is used only on serial channels. Use the <i>rate</i> argument to specify the frequency with which the control leads are sampled, in samples per second. Default is 0. <p>Note Control lead update packets are independent of the data packets from the same channel.</p>

Command or Action	Purpose
<p>Step 7 <code>control-lead state {active fail} output-lead{on off follow} [{local remote} input-lead]</code></p> <p>Example:</p> <pre>Router(config-cem)# control-lead state active rts follow remote cts</pre>	<p>(Optional) Specifies the state of each output control lead on a serial CEM channel.</p> <ul style="list-style-type: none"> • This command is used only on serial channels. • Use the active keyword to specify the state of the control lead when the connection is active. • Use the fail keyword to specify the state of the control lead when the connection has failed. • Use the <i>output-lead</i> argument to specify the name of the control lead. • Use the on keyword to specify that the control lead is permanently asserted. • Use the off keyword to specify that the control lead is permanently not asserted. • Use the follow keyword to specify that the control lead is to follow any changes in the state of an input control lead specified by the local or remote keywords and the <i>input-lead</i> argument. • Use the <i>input-lead</i> argument to specify the name of the local or remote control lead to follow. <p>Note Control lead update packets are independent of the data packets for the same channel.</p> <p>Note The control-lead sampling-rate parameter must be set to non-zero for this feature to operate.</p>
<p>Step 8 <code>data-strobe input-lead {on off}</code></p> <p>Example:</p> <pre>Router(config-cem)# data-strobe dtr on</pre>	<p>(Optional) Specifies that an input control lead is to be monitored and data is packetized and sent only when the specified control lead is in the specified state.</p> <ul style="list-style-type: none"> • This command is used only on serial channels. • Use the <i>input-lead</i> argument to specify the input control lead to be monitored to determine whether input data is to be packetized. • Use the on keyword to specify that data packets are to be sent from this CEM channel only when the specified input lead is asserted. • Use the off keyword to specify that data packets are to be sent from this CEM channel only when the specified input lead is not asserted. • Use this command to save bandwidth when the attached CPE is inactive. <p>Note Control lead update packets are still sent even if data packets are withheld.</p>

Command or Action	Purpose
<p>Step 9 Cisco NM-CEM-4SER:</p> <p>Example:</p> <pre> idle-pattern length pattern1 [<i>pattern2</i>] </pre> <p>Example:</p> <pre> Cisco NM- CEM-4TE1: </pre> <p>Example:</p> <pre> idle-pattern pattern1 </pre> <p>Example:</p> <pre> Cisco NM- CEM-4SER: </pre> <p>Example:</p> <pre> Router(config-cem)# idle- pattern 53 0x12345678 0x87654321 </pre> <p>Example:</p> <pre> Cisco NM- CEM-4TE1: </pre>	<p>(Optional) Defines the idle data pattern to send to the attached CPE when packets are lost or the de-jitter buffer experiences an under-run condition.</p> <p>For serial CEM channels:</p> <ul style="list-style-type: none"> • A bit pattern up to 64 bits long may be specified. • Use the <i>pattern1</i> argument to specify up to 32 bits of the least significant bits of the idle data pattern, in hex notation. Default is 0xFF. • Use the <i>pattern2</i> argument to specify the most significant bits of the idle data pattern, in hex notation. If the <i>length</i> argument is 32 bits or less, this argument is not permitted. • Use the <i>length</i> argument to specify the total length of the repeating bit pattern. Default is 8 bits. <p>For T1 or E1 CEM channels:</p> <ul style="list-style-type: none"> • An eight-bit pattern is specified.

Command or Action	Purpose
<p>Example:</p> <pre>Router(config-cem)# idle-pattern 0x66</pre>	
<p>Step 10 signaling</p> <p>Example:</p> <pre>Router(config-cem)# signaling</pre>	<p>(Optional) Enables the transport of Channel Associated Signaling (CAS) bits.</p> <p>Note This command applies only to framed T1 or E1 data channels.</p>
<p>Step 11 payload-compression</p> <p>Example:</p> <pre>Router(config-cem)# payload-compression</pre>	<p>(Optional) Enables payload compression on a CEM channel.</p> <p>Note Enabling payload compression adds a delay equal to one packet time.</p>
<p>Step 12 data-protection</p> <p>Example:</p> <pre>Router(config-cem)# data-protection</pre>	<p>(Optional) Enables data protection by transmitting each data bit twice, once in each of two consecutive data packets.</p> <ul style="list-style-type: none"> Use the data-protection command to protect transmissions from the effects of lost IP packets. <p>Caution Use this command carefully because it increases the network bandwidth used by the CEM connection.</p>
<p>Step 13 ip dscp dscp</p> <p>Example:</p> <pre>Router(config-cem)# ip dscp 36</pre>	<p>(Optional) Configures the IP Differentiated Service Code Point (DSCP) for packets originating from this CEM channel.</p> <ul style="list-style-type: none"> Use the <i>dscp</i> argument to specify the value placed in the DSCP field of IP packets originating from this channel. Default is 46. <p>Note If DSCP is configured, the ip tos and ip precedence commands are not available because they are mutually exclusive.</p>
<p>Step 14 ip tos tos</p> <p>Example:</p> <pre>Router(config-cem)# ip tos 11</pre>	<p>(Optional) Configures the IP type of service (ToS) bits for the CEM channel.</p> <ul style="list-style-type: none"> Use the <i>tos</i> argument to specify the value placed in the ToS field of IP packets originating from this channel. Default is 5. <p>Note If DSCP is configured, the ip tos command is not available because they are mutually exclusive.</p>

Command or Action	Purpose
<p>Step 15 <code>ip precedence precedence</code></p> <p>Example:</p> <pre>Router(config-cem)# ip precedence 7</pre>	<p>(Optional) Configures the IP precedence bits for the CEM channel.</p> <ul style="list-style-type: none"> Use the <i>precedence</i> argument to specify the value placed in the precedence field of IP packets originating from this channel. Default is 0. <p>Note If DSCP is configured, the ip precedence command is not available because they are mutually exclusive.</p>
<p>Step 16 <code>loopback {local network}</code></p> <p>Example:</p> <pre>Router(config-cem)# loopback network</pre>	<p>(Optional) Creates a loopback from a CEM serial channel.</p> <ul style="list-style-type: none"> Use the local keyword to create a loopback where the information from a locally-attached CPE is transmitted back to the locally-attached CPE. Use the network keyword to create a loopback where the data received over the network from a remotely-attached CPE is transmitted back to the remotely-attached CPE. <p>Note For configuring a loopback on a T1 or E1 port, see the Creating CEM Channels on the T1 E1 Line, page 170.</p>
<p>Step 17 <code>exit</code></p> <p>Example:</p> <pre>Router(config-cem)# exit</pre>	<p>Exits CEM configuration mode and returns to global configuration mode.</p> <ul style="list-style-type: none"> Use this command one more time to exit to privileged EXEC mode.

- [What to Do Next, page 179](#)

What to Do Next

Proceed to the [Configuration Examples for CEoIP, page 179](#).

Configuration Examples for CEoIP

- [Configuring a T1 CEM Network Module Example, page 179](#)

Configuring a T1 CEM Network Module Example

The following example shows a basic configuration of a T1 network module to configure the CEoIP feature.

```
card type t1 0
controller t1 4/0
cem-group 6 timeslots 1-4,9,10 speed 64
framing esf
linecode b8zs
clock source adaptive 6
cablelength long -15db
crc-threshold 512
description T1 line to 3rd floor PBX
```

```

loopback network
no shutdown
exit
cem 2/1/6
xconnect 10.2.0.1 0 encapsulation udp
local ip-address 10.2.0.9
local udp port 15901
remote udp port 15901
payload-size 512
dejitter-buffer 80
signaling
exit

```

Additional References

For additional information related to the CEoIP feature, refer to the following references:

Related Documents

Related Topic	Document Title
CEoIP NMs	<i>Release Notes for Cisco NM-CEM-4TE1 and NM-CEM-4SER Network Module Software</i>

Standards

Standards	Title
GR-1089	<i>Electromagnetic Compatibility and Electrical Safety - Generic Criteria for Network Telecommunications Equipment</i>
GR-63	Network Equipment-Building System (NEBS) Requirements: Physical Protection
TIA/EIA-IS-968	Technical Requirements for Connection of Terminal Equipment to the Telephone Network

MIBs

MIBs	MIBs Link
<ul style="list-style-type: none"> • OLD-CISCO-CHASSIS-MIB • RFC1406-MIB • CISCO-ENTITY-VENDORTYPE-OID-MIB 	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 1406	<i>Definitions of Managed Objects for the DS1 and E1 Interface Types</i>

RFCs	Title
RFC 2495	Definitions of Managed Objects for the DS1, E1, DS2 and E2 Interface Types
	Note CEoIP supports RFC2495 to the same extent as IOS supports this RFC.

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

Command Reference

The following commands are introduced or modified in the feature or features documented in this module. For information about these commands, see the *Cisco IOS Interface and Hardware Component Command Reference* at http://www.cisco.com/en/US/docs/ios/interface/command/reference/ir_book.html. For information about all Cisco IOS commands, go to the Command Lookup Tool at <http://tools.cisco.com/Support/CLILookup> or to the *Cisco IOS Master Commands List* .

- **cem**
- **cem-group**
- **clear cem**
- **clock mode**
- **clock source (CEM)**
- clock-switchover
- **control-lead sampling-rate**
- **control-lead state**
- **crc-threshold**
- **data-protection**
- **data-strobe**
- **default (CEM)**
- **dejitter-buffer**
- emulation-mode
- **framing (CEM)**
- **idle-pattern**
- **ip dscp**
- **local ip address**
- **local udp port**
- **loopback (CEM)**
- **payload-compression**

- **payload-size**
- **remote udp port**
- **show cem**
- **signaling**
- **xconnect (CEM)**

Feature Information for Circuit Emulation over IP

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Release	Modification
12.3(7)T	This feature was introduced.
12.2(33)SRD	Included information on clock switchover and unidirectional emulation mode.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source

The Lossless Compression R1, ATM Cell Switching, and External BITS Clocking Source features introduce a new compression technique in DSP firmware and add enhancements to Cisco IOS that include cell switching on ATM segmentation and reassembly (SAR), and the use of an external BITS clocking source. These features enable Cisco multiservice routers to be used to transparently groom and compress traffic in a wireless service provider network and enable a service provider to optimize the bandwidth used to backhaul the traffic from a cell site to the mobile central office for more efficient use of existing T1 and E1 lines.

Feature Specifications for Cisco Lossless Compression R1, ATM Cell Switching, and External BITS Clocking Source

Feature History

Release	Modification
12.3(4)XD	These features were introduced.
12.3(7)T	These features were integrated into Cisco IOS Release 12.3(7)T.

Supported Platforms

Cisco 3660, Cisco 3745

- [Finding Feature Information, page 184](#)
- [Prerequisites for Cisco Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source, page 184](#)
- [Restrictions for Cisco Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source, page 184](#)
- [Information About Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source, page 185](#)
- [How to Configure Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source, page 186](#)
- [Verifying Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source Configuration, page 198](#)
- [Additional References, page 200](#)
- [Feature Information for Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source, page 201](#)

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.

Prerequisites for Cisco Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source

The Lossless Compression R1, ATM Cell Switching, and External BITS Clocking Source features require a Cisco 3660 or Cisco 3745 with the following components installed:

Table 11 Supported Network Modules

Feature	Cisco 3660	Cisco 3745
Lossless compression R1	NM-HDV	NM-HDV
ATM cell switching	AIM-ATM or AIM-ATM-VOICE-30 NM- <i>x</i> FE2W with VWIC- <i>x</i> MFT-T1/E1	AIM-ATM or AIM-ATM-VOICE-30 NM- <i>x</i> FE2W with VWIC- <i>x</i> MFT-T1/E1 VWIC- <i>x</i> MFT-T1/E1 (on-board WIC slot)
BITS clocking	NM-HDV NM- <i>x</i> FE2W with VWIC- <i>x</i> MFT-T1/E1	NM-HDV NM- <i>x</i> FE2W with VWIC- <i>x</i> MFT-T1/E1 VWIC- <i>x</i> MFT-T1/E1 (on-board WIC slot)

Restrictions for Cisco Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source

- Operations, administration, and maintenance (OAM) cell insertion is not supported on cell-switched PVCs.
- AIM-ATM and AIM-ATM-VOICE-30 modules support a maximum of four T1/E1s. This can consist of two incoming and two outgoing, or three incoming and one outgoing T1/E1s. An IMA group cannot be split between multiple AIMS.

- Certain combinations of AIM modules can become inoperable when installed in a Cisco 3745. This problem only affects Cisco 3745 routers manufactured before June 11, 2003. See the following field notice for detailed information about this problem:

http://www-tac.cisco.com/Support_Library/field_alerts/fn25194.html

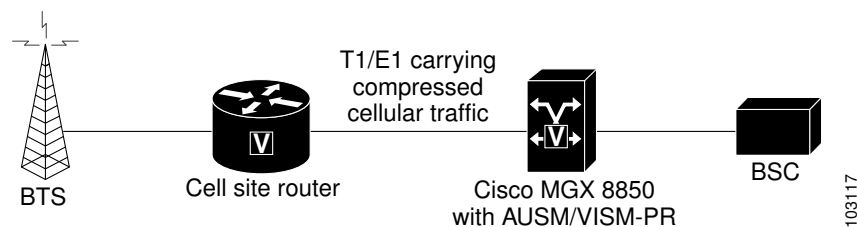
- Voice activity detection (VAD) and echo cancellation are disabled when lossless compression is enabled.
- Lossless compression R1 is supported for VoATM calls with AAL2 and subcell multiplexing. VoIP calls are not supported at this time.
- ATM cell switching is limited to a maximum of 25 connections per AIM-ATM.
- Do not configure more than 29 LLCC channels per NM-HDV module. Configuring more than 29 LLCC channels can cause unreliable operation.
- J1 controller is not supported.
- Traffic policing is not supported.
- For Cisco 3660 routers with two NM-HDV modules installed, do not install the modules in the following slot combinations:
 - Slot 1 and Slot 3
 - Slot 2 and Slot 4
 - Slot 5 and Slot 6

Using these slot combinations can result in packet loss.

Information About Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source

The Lossless Compression R1, ATM Cell Switching, and External BITS Clocking Source features work together to groom and compress T1 and E1 traffic between cell sites and a mobile central office. These features require a Cisco 3660 or Cisco 3745 router to be installed at the base transceiver station (BTS). This cell site router performs ATM switching and compression of cell site traffic for transport to the base station controller (BSC). A Cisco MGX 8850 with AUSM and VISM-PR terminates the T1/E1 lines that carry lossless compression codec (LLCC) traffic, converting the traffic back to PCM before passing it to the BSC. The figure below shows a sample topology that makes use of the Lossless Compression R1, ATM Cell Switching, and External BITS Clocking Source features.

Figure 14 Lossless Compression R1, ATM Cell Switching, and External BITS Clocking Source Features



- [Lossless Compression Codec on NM-HDV](#), page 186
- [ATM Cell Switching on AIM-ATM and AIM-ATM-VOICE-30](#), page 186
- [BITS Clocking on the Cisco 3660 and Cisco 3745](#), page 186

Lossless Compression Codec on NM-HDV

The Lossless Compression R1 feature introduces a new compression technique in DSP firmware and the VISM card-- the lossless compression codec (LLCC). LLCC operates in a similar fashion to the existing clear channel codec: the decoded 64kbps PCM stream is a bit-exact replica of the PCM stream provided on the TDM side of the encoding DSP. However, rather than simply packetizing the PCM stream, the LLCC encoder applies a lossless data compression scheme. This results in a net reduction in the data transmission rate, yielding a reduction in the packet transmission rate.

ATM Cell Switching on AIM-ATM and AIM-ATM-VOICE-30

The Cisco ATM Cell Switching feature enables the router to perform cell switching between two ATM connections on AIM-ATM and AIM-ATM-VOICE-30 cards, giving the router the ability to receive ATM traffic from the BTS and backhaul it to the mobile central office.

BITS Clocking on the Cisco 3660 and Cisco 3745

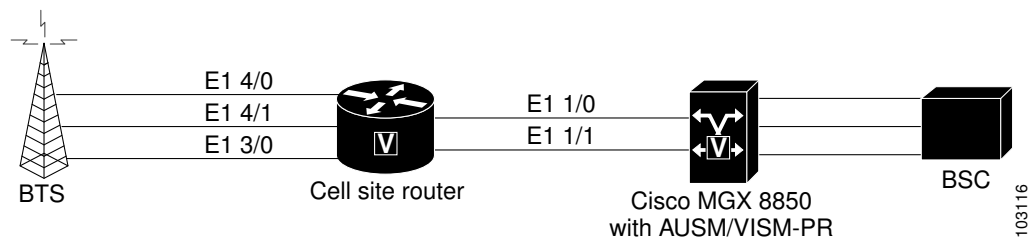
BITS (Building Integrated Timing Supply) network clocking enables a Cisco 3660 or Cisco 3745 router to derive network timing from the central office. BITS must be configured on the cell site router to support this feature.

How to Configure Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source

The procedures for configuring the Lossless Compression R1, ATM Cell Switching, and External BITS Clocking Source features require the following tasks:

The instructions that follow refer to the sample configuration shown in the figure below. With this configuration, the cell site router supports three E1 connections to the BTS. Compressed cellular traffic is transported to the BSC (by way of the Cisco MGX 8850) over the E1 1/0 and E1 1/1 interfaces. Additionally, BITS clocking is derived from E1 1/1.

Figure 15 **Sample Configuration**



- [Configuring the Cell Site Router for BITS Clocking, page 187](#)
- [Configuring ATM Cell Switching, page 188](#)
- [Configuring the Lossless Compression Codec, page 191](#)
- [Disabling Connection Admission Control, page 196](#)

Configuring the Cell Site Router for BITS Clocking

BITS clocking enables the router at a cell site to derive timing from the mobile central office. BITS clocking ensures that data flows to a single network clock source, preventing mismatches and data slips in traffic between the BTS and the BSC. The procedure that follows configures the AIM to receive BITS clocking from E1 1/1 controller.

Summary Steps

- 1 **enable**
- 2 **configure terminal**
- 3 **network-clock-participate** *slot number*
- 4 **network-clock-select** **priority** *slot number*
- 5 **controller e1** *slot/port*
- 6 **clock source** {**line** [**primary** | **bits**] | **internal**}

Detailed Steps

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **network-clock-participate** **slot** *number*
4. **network-clock-select** **priority** *slot number*
5. **controller t1** | **e1** *slot/port*
6. **clock source** { **line** [**primary** | **bits**] | **internal** }

DETAILED STEPS

Command or Action	Purpose
Step 1 enable Example: Router> enable	Enables privileged EXEC mode. Enter your password when prompted.
Step 2 configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3 network-clock-participate <i>slot number</i> Example: Router(config)# network-clock-participate slot 1	Allows the network module in the specified slot to use the network clock for its timing.

Command or Action	Purpose
Step 4 <code>network-clock-select priority slot number</code> Example: <pre>Router(config)# network-clock-select 1 E1 1/1</pre>	Specifies a port to be used as a timing source for the network clock, and the priority level for the use of that port. The source that is given the highest priority is used first; if it becomes unavailable, the source with the second-highest priority is used, and so forth.
Step 5 <code>controller t1 e1 slot/port</code> Example: <pre>Router(config)# controller e1 1/1</pre>	Enters controller configuration mode for the selected T1 or E1.
Step 6 <code>clock source { line [primary bits] internal}</code> Example: <pre>Router(config-controller)# clock source line bits</pre>	Specifies that the clock is generated from the T1 or E1 BITS source.

Configuring ATM Cell Switching

The procedure that follows configures the cell site router to switch ATM traffic with the Cisco MGX 8850 at the BSC. This procedure configures ATM switching between E1 3/0 and E1 1/0, using the AIM installed in Slot 1.

Summary Steps

- 1 `enable`
- 2 `configure terminal`
- 3 `network-clock-participate slot number`
- 4 `network-clock-participate slot number`
- 5 `network-clock-participate aim number`
- 6 `controller t1 | e1 slot /port`
- 7 `mode atm aim aim-slot`
- 8 `controller t1 | e1 slot/port`
- 9 `mode atm aim aim-slot`
- 10 `interface atm interface-number /subinterface-number`
- 11 `pvc vpi /vci l2transport`
- 12 `interface atm interface-number /subinterface-number`
- 13 `pvc vpi /vci l2transport`
- 14 `connect id atm slot /port-1 atm slot/port-2`

Detailed Steps

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **network-clock-participate slot *number***
4. **network-clock-participate slot *number***
5. **network-clock-participate aim *number***
6. **controller t1 | e1 slot/port**
7. **mode atm aim *aim-slot***
8. **controller t1 | e1 slot/port**
9. **mode atm aim *aim-slot***
10. **interface atm *interface-number* /*subinterface-number***
11. **pvc *vpi* /*vci* l2transport**
12. **interface atm *interface-number* /*subinterface-number***
13. **pvc *vpi* /*vci* l2transport**
14. **connect id atm *slot* /*port-1* atm *slot* /*port-2***

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. Enter your password when prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	network-clock-participate slot <i>number</i> Example: Router(config)# network-clock-participate slot 1	Enables the network module in the specified slot to use the network clock for its timing.
Step 4	network-clock-participate slot <i>number</i> Example: Router(config)# network-clock-participate slot 3	Enables the network module in the specified slot to use the network clock for its timing.

	Command or Action	Purpose
Step 5	<p>network-clock-participate aim <i>number</i></p> <p>Example:</p> <pre>Router(config)# network-clock-participate aim 0</pre>	Specifies that the AIM in Slot 0 will derive clocking from the network source.
Step 6	<p>controller t1 e1 slot/port</p> <p>Example:</p> <pre>Router(config)# controller e1 1/0</pre>	Enters controller configuration mode for the selected T1 or E1.
Step 7	<p>mode atm aim <i>aim-slot</i>:</p> <p>Example:</p> <pre>Router(config-controller)# mode atm aim 0</pre>	Sets the mode of the T1 or E1 controller in AIM Slot 0.
Step 8	<p>controller t1 e1 slot/port</p> <p>Example:</p> <pre>Router(config)# controller e1 3/0</pre>	Enters controller configuration mode for the selected T1 or E1.
Step 9	<p>mode atm aim <i>aim-slot</i></p> <p>Example:</p> <pre>Router(config-controller)# mode atm aim 0</pre>	Sets the mode of the T1 or E1 controller in AIM Slot 0.
Step 10	<p>interface atm <i>interface-number /subinterface-number</i></p> <p>Example:</p> <pre>Router(config) # interface atm 1/0</pre>	Enters configuration mode for the selected ATM interface.
Step 11	<p>pvc <i>vpi /vci</i> l2transport</p> <p>Example:</p> <pre>Router(config-if)# pvc 10/110 l2transport</pre>	Creates a PVC for the virtual path identifier (VPI) and virtual channel identifier (VCI) and specifies that the PVC is switched, not terminated.

Command or Action	Purpose
<p>Step 12 <code>interface atm <i>interface-number</i> /<i>subinterface-number</i></code></p> <p>Example:</p> <pre>Router (config) # interface atm 3/0</pre>	<p>Enters configuration mode for the selected ATM interface.</p>
<p>Step 13 <code>pvc <i>vpi</i> /<i>vci</i> l2transport</code></p> <p>Example:</p> <pre>Router(config-if)# pvc 30/130 l2transport</pre>	<p>Creates a PVC for the VPI and VCI and specifies that the PVC is switched.</p>
<p>Step 14 <code>connect id atm <i>slot</i> /<i>port-1</i> atm <i>slot</i> /<i>port-2</i></code></p> <p>Example:</p> <p>Example:</p> <pre>Router(config)# connect Switched-Conn atm 1/0 10/110 atm 3/0 30/130</pre>	<p>Defines connections between T1 or E1 controller ports and the ATM interface.</p>

Configuring the Lossless Compression Codec

The procedure that follows configures an LLCC voice channel on E1 4/0 and sends it over the ATM network using E1 1/0 and the AIM installed in Slot 1.

Summary Steps

- 1 **enable**
- 2 **configure terminal**
- 3 **network-clock-participate slot *number***
- 4 **network-clock-participate slot *number***
- 5 **network-clock-participate aim *number***
- 6 **voice service {pots | voatm | vofr | voip}**
- 7 **session protocol aal2**
- 8 **subcell-mux**
- 9 **codec aal2-profile custom *profile-number* codec**
- 10 **controller t1 | e1 *slot/port***
- 11 **mode atm aim *aim-slot***
- 12 **controller t1 | e1 *slot/port***
- 13 **ds0-group *ds0-group-number* timeslots *timeslot-list* type *signaling method***
- 14 **interface atm *interface-number* /*subinterface-number***
- 15 **pvc *vpi* /*vci***
- 16 **vbr-rt *peak-rate* *average-rate* *burst***

- 17 **encapsulation aal2**
- 18 **dial-peer voice tag voatm**
- 19 **destination-pattern string**
- 20 **session protocol aal2-trunk**
- 21 **session target interface pvc vpi/vci**
- 22 **signal-type cas | cept | ext-signal | transparent**
- 23 **codec aal2-profile custom profile-number codec**
- 24 **voice-port {slot-number/subunit-number/port | slot/port:ds0-group-no}**
- 25 **playout-delay {fax | maximum | nominal} milliseconds**
- 26 **connection {plar | tie-line | plar-opx} digits [{trunk digits [answer-mode]}**

Detailed Steps

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **network-clock-participate slot number**
4. **network-clock-participate slot number**
5. **network-clock-participate aim number**
6. **voice service {pots | voatm | vofr | voip}**
7. **session protocol aal2**
8. **subcell-mux**
9. **codec aal2-profile custom profile-number codec**
10. **controller t1 | e1 slot/port**
11. **mode atm aim aim-slot**
12. **controller t1 | e1 slot/port**
13. **ds0-group ds0-group-number timeslots timeslot-list type signaling method**
14. **interface atm interface-number /subinterface-number**
15. **pvc vpi /vci**
16. **vbr-rt peak-rate average-rate burst**
17. **encapsulation aal2**
18. **dial-peer voice tag voatm**
19. **destination-pattern string**
20. **session protocol aal2-trunk**
21. **session target interface pvc vpi/vci**
22. **signal-type cas | cept | ext-signal | transparent**
23. **codec aal2-profile custom profile-number codec**
24. **voice-port {slot-number/subunit-number/port | slot/port:ds0-group-no}**
25. **playout-delay {fax | maximum | nominal} milliseconds**
26. **connection {plar | tie-line | plar-opx} digits [{trunk digits [answer-mode]}**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Router> enable</pre>	Enables privileged EXEC mode. Enter your password when prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>network-clock-participate slot <i>number</i></p> <p>Example:</p> <pre>Router(config)# network-clock-participate slot 1</pre>	Enables the network module in the specified slot to use the network clock for its timing.
Step 4	<p>network-clock-participate slot <i>number</i></p> <p>Example:</p> <pre>Router(config)# network-clock-participate slot 4</pre>	Enables the network module in the specified slot to use the network clock for its timing.
Step 5	<p>network-clock-participate aim <i>number</i></p> <p>Example:</p> <pre>Router(config)# network-clock-participate aim 0</pre>	Specifies that the AIM in Slot 0 will derive clocking from the network source.
Step 6	<p>voice service {pots voatm vofr voip}</p> <p>Example:</p> <pre>Router(config)# voice service voatm</pre>	Enters voice service configuration mode and specifies VoATM as the encapsulation type.
Step 7	<p>session protocol aal2</p> <p>Example:</p> <pre>Router(config-voi-serv)# session protocol aal2</pre>	Enters voice-service-session configuration mode and specifies ATM adaptation layer 2 (AAL2) trunking.

	Command or Action	Purpose
Step 8	<p>subcell-mux</p> <p>Example:</p> <pre>Router(conf-voi-serv-sess)# subcell-mux</pre>	Enables AAL2 common part sublayer (CPS) subcell multiplexing.
Step 9	<p>codec aal2-profile custom <i>profile-number</i> codec</p> <p>Example:</p> <pre>Router# codec aal2-profile custom 51 0 0 llcc 40 0 15</pre>	Sets the codec profile for the DSP on a per-call basis and specifies the lossless compression codec.
Step 10	<p>controller t1 e1 slot/port</p> <p>Example:</p> <pre>Router(config)# controller e1 1/0</pre>	Enters controller configuration mode for the selected T1 or E1.
Step 11	<p>mode atm aim <i>aim-slot</i></p> <p>Example:</p> <pre>Router(config-controller)# mode atm aim 0</pre>	Sets the mode of the T1 or E1 controller in AIM Slot 0.
Step 12	<p>controller t1 e1 slot/port</p> <p>Example:</p> <pre>Router(config)# controller e1 4/0</pre>	Enters controller configuration mode for the selected T1 or E1.
Step 13	<p>ds0-group <i>ds0-group-number</i> timeslots <i>timeslot-list</i> type <i>signaling method</i></p> <p>Example:</p> <pre>Router(config-controller)# ds0-group 0 timeslots 1 type ext-sig</pre>	Specifies the DS0 time slots that make up a logical voice port on a T1 or E1 controller and specifies the signaling type used by the router.
Step 14	<p>interface atm <i>interface-number</i> /<i>subinterface-number</i></p> <p>Example:</p> <pre>Router(config) # interface atm 1/0</pre>	Enters configuration mode for the selected ATM interface.

Command or Action	Purpose
<p>Step 15 <code>pvc vpi /vci</code></p> <p>Example:</p> <pre>Router(config-if-atm)# pvc 10/110</pre>	<p>Enters configuration mode for the selected PVC.</p>
<p>Step 16 <code>vbr-rt peak-rate average-rate burst</code></p> <p>Example:</p> <pre>Router(config-if-atm-pvc)# vbr-rt 1920 1920 255</pre>	<p>Configures real-time variable bit rate (VBR) for VoATM voice connections.</p>
<p>Step 17 <code>encapsulation aal2</code></p> <p>Example:</p> <pre>Router(config-if-atm-pvc)# encapsulation aal2</pre>	<p>Configures the encapsulation type for the ATM virtual circuit.</p>
<p>Step 18 <code>dial-peer voice tag voatm</code></p> <p>Example:</p> <pre>Router(config)# dial-peer voice 1001 voatm</pre>	<p>Defines a dial-peer and specifies the method of voice encapsulation as VoATM.</p>
<p>Step 19 <code>destination-pattern string</code></p> <p>Example:</p> <pre>Router(config-dial-peer)# destination-pattern 1001</pre>	<p>Specifies the prefix to be used by the dial peer.</p>
<p>Step 20 <code>session protocol aal2-trunk</code></p> <p>Example:</p> <pre>Router(config-dial-peer)# session protocol aal2-trunk</pre>	<p>Specifies the dial peer uses AAL2 nonswitched trunk session protocol.</p>
<p>Step 21 <code>session target interface pvc vpi/vci</code></p> <p>Example:</p> <pre>Router(config-dial-peer)# session target atm 1/0 pvc 10/100 9</pre>	<p>Specifies the network-specific address for the VoATM dial peer.</p>

Command or Action	Purpose
Step 22 <code>signal-type cas cept ext-signal transparent</code> Example: <code>Router(config-dial-peer)# signal-type ext-signal</code>	Specifies that external signaling is used when connecting to the dial peer. The DSP does not generate any signaling frames.
Step 23 <code>codec aal2-profile custom profile-number codec</code> Example: <code>Router(config-dial-peer)# codec aal2-profile custom 51 llcc</code>	Sets the codec profile for the DSP on a per-call basis and specifies the lossless compression codec.
Step 24 <code>voice-port {slot-number/subunit-number/port slot/port:ds0-group-no}</code> Example: <code>Router(config)# voice-port 2/0:0</code>	Enters voice-port configuration mode.
Step 25 <code>playout-delay {fax maximum nominal} milliseconds</code> Example: <code>Router(config-voice-port)# playout-delay nominal 25</code>	Tunes the playout buffer to accommodate packet jitter caused by switches in the WAN. The nominal keyword specifies the initial (and minimum allowed) delay time that the DSP inserts before playing out voice packets, in milliseconds.
Step 26 <code>connection {plar tie-line plar-opx} digits {trunk digits [answer-mode]}</code> Example: <code>Router(config-voice-port)# connection trunk 1001</code>	Associates this voice-port to destination-pattern 1001.

**Note**

To ensure that the voice-port configuration takes affect, issue the **shutdown** command, followed by **no shutdown** to enable it again.

Disabling Connection Admission Control

Connection admission control (CAC) is a set of actions taken by each ATM switch during connection setup to determine whether the requested QoS will violate the QoS guarantees for established connections. CAC reserves bandwidth for voice calls, however, the bandwidth required when LLCC is used is dynamic and usually less than what is generally reserved by CAC. Disabling CAC may help in better utilization of bandwidth when LLCC is used. The procedure that follows disables CAC.

Summary Steps

- 1 **enable**
- 2 **configure terminal**
- 3 **interface atm** *interface-number/subinterface-number*
- 4 **pvc** *vpi/vci*
- 5 **cac_off**

- [Detailed Steps, page 197](#)

Detailed Steps**SUMMARY STEPS**

1. **enable**
2. **configure terminal**
3. **interface atm** *interface-number /subinterface-number*
4. **pvc** *vpi /vci*
5. **cac_off**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. Enter your password when prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	interface atm <i>interface-number /subinterface-number</i> Example: Router(config) # interface atm 1/0	Enters configuration mode for the selected ATM interface.
Step 4	pvc <i>vpi /vci</i> Example: Router(config-if-atm)# pvc 10/110	Enters configuration mode for the selected PVC.

Command or Action	Purpose
Step 5 <code>cac_off</code> Example: Router# (config-if-atm-vc)# <code>cac_off</code>	Disables call admission control.

Verifying Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source Configuration

This section provides a set of **show** commands you can use to verify the configuration of the Lossless Compression R1, ATM Cell Switching, and External BITS Clocking Source features. It includes the following commands:

show connection all

The following example shows output from the **show connection all** command. In this example, Switched-Conn is a cell-switched connection established between PVC 10/110 and PVC 30/130, which are configured under ATM1/0 and ATM3/0 respectively.

```
Router# show connection all
ID      Name          Segment 1          Segment 2          State
-----
3      V-100-700      E1 1/0(VOICE) 00    DSP 07/00/00      UP
4      V-120-700      E1 1/2(VOICE) 00    DSP 07/00/00      UP
5      Switched-Conn  ATM1/0 10/110      ATM3/0 30/130      UP
```

The **show connection all** command displays the state of Switched-Conn. If it is in the UP state, then it means the ATM cell switching connection is operational.

show voice dsp

The following example shows output from the **show voice dsp** command:

```
Router# show voice dsp
DSP  DSP          DSPWARE CURR  BOOT          PAK  TX/RX
TYPE NUM CH CODEC  VERSION STATE  STATE  RST AI VOICEPORT TS ABORT PACK COUNT
==== == == =====
C549 000 04 llcc      4.3.392 busy  idle      0 4/0:0 04 0 1752/1752
```

The **show voice dsp** command shows if the LLCC codec has been applied to the voice port. Additionally, the TX/RX COUNT indicates if packet exchange is occurring. If LLCC is operational, then TX/RX COUNT will display similar values.

show voice call port-id

The **show voice call** command gives detailed information about the lossless compression codec. The following example shows output from the **show voice call** command:

**Note**

The **show voice call** command has a limitation that causes it to display invalid values. To ensure that accurate values are reported, invoke this command twice and look at the second output.

```
Router# show voice call 4/0:0
4/0:0 1
    vtsp level 0 state = S_CONNECTvpm level 1 state = S_TRUNKED
vpm level 0 state = S_UP
lossless compression summary:
  average compression ratio since reset      = 50
  current compression ratio                 = 50
  max buffer size (ms)                     = 41
  nominal buffer size (ms)                 = 25
  current buffer size (ms)                 = 26
  total encoder input frame count           = 5534
  total encoder output frame count          = 2767
  encoded tx front-end compressed frame count = 2767
  encoded tx back-end compressed frame count = 0
  encoded tx frame count (no compression)   = 0
  underflow error count                     = 0
  overflow error count                      = 0
  decode error count                       = 0
  tx signalling frame count                 = 11
  rx signalling frame count                 = 10
  rx bad checksum frame count               = 0
  rx good checksum frame count              = 2777
```

show voice trunk supervisory summary

The following example shows output from the **show voice trunk supervisory summary** command:

```
Router# show voice trunk supervisory summary
SLOW SCAN
4/0:0(1) : state : TRUNK_SC_CCS_CONNECT, master
```

show interfaces

The following example shows output from the **show interfaces** command:

```
Router# show interfaces atm1/0
ATM1/0 is up, line protocol is up
  Hardware is ATM AIM E1
  MTU 4470 bytes, sub MTU 4470, BW 1920 Kbit, DLY 20000 usec,
    reliability 0/255, txload 1/255, rxload 1/255
  Encapsulation ATM, loopback not set
  Encapsulation(s): AAL5
  255 maximum active VCs, 256 VCs per VP, 0 current VCCs
  VC Auto Creation Disabled.
  VC idle disconnect time: 300 seconds
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: Per VC Queueing
  30 second input rate 0 bits/sec, 0 packets/sec
  30 second output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 no buffer
  Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  0 packets output, 0 bytes, 0 underruns
  0 output errors, 0 collisions, 1 interface resets
  0 output buffer failures, 0 output buffers swapped out
```

Additional References

Related Documents

Related Topic	Document Title
Configuring voice features	<i>Cisco IOS Voice Configuration Library</i>
Configuring ATM advanced integration modules	AIM-ATM and AIM-ATM-VOICE-30 on the Cisco 2600 Series, Cisco 3660, and Cisco 3700 Series
Configuring high-density voice network modules	Digital E1 Packet Voice Trunk Network Module Interfaces

Standards

Standards ¹⁷	Title
No new standards are supported by this feature.	

MIBs

MIBs	MIBs Link
<ul style="list-style-type: none"> No new MIBs are supported by this feature. CISCO-VOICE-COMMON-DIAL-CONTROL-MIB was modified. 	<p>To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:</p> <p>http://www.cisco.com/go/mibs</p>

RFCs

RFCs ¹⁸	Title
No new RFCs are supported by this feature.	

¹⁷ Not all supported standards are listed.

¹⁸ Not all supported RFCs are listed.

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Lossless Compression R1 ATM Cell Switching and External BITS Clocking Source

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 12 Feature Information for Phrase Based on Module Title

Feature Name	Releases	Feature Information
Lossless Compression R1, ATM Cell Switching, and External BITS Clocking Source	12.3(4)XD	These features were introduced.
Lossless Compression R1, ATM Cell Switching, and External BITS Clocking Source	12.3(7)T	These features were integrated into Cisco IOS Release 12.3(7)T.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



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.



Minimal Disruptive Restart of VIP Cards

The Minimal Disruptive Restart (MDR) of VIP Cards feature optimizes the reload time of a VIP card on a Cisco 7500 series router after a software failure has occurred. The amount of time for a VIP card to reload with the MDR functionality varies depending on the port adapter in the VIP card. With this software enhancement, the reload time of a VIP card with the Single Line Card Reload (SLCR) technology is decreased from approximately 30 seconds to approximately 5 seconds. This improvement provides high availability and minimizes equipment downtime. As an additional part of this feature, users can initiate a VIP card reload with the MDR functionality on a single slot using the **microcode reload** command.

The MDR functionality has the following limitations:

- If a VIP card software failure occurs within 5 minutes of up time, the VIP card is not reloaded with the MDR functionality. Instead, a standard reset and microcode reload is performed. If a software failure occurs after 5 minutes of up time, the VIP card is reloaded with the MDR functionality.
- A VIP card can be reloaded with the MDR functionality five consecutive times. If a VIP card must be reloaded for a sixth consecutive time, it is not reloaded with the MDR functionality. Instead, a standard reset and microcode reload is performed.

These limitations are not applicable when performing a VIP card reload using the **microcode reload** command. There is no limit to the number of times a VIP card can be reloaded with this command.

Feature History for the Minimal Disruptive Reload of VIP Cards Feature

Release	Modification
12.3(8)T	This feature was introduced.



Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.





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 209](#)
- [Restrictions for Rate Based Satellite Control Protocol, page 209](#)
- [Information About Rate Based Satellite Control Protocol, page 209](#)
- [How to Configure Rate Based Satellite Control Protocol, page 213](#)
- [Configuration Examples for Rate Based Satellite Control Protocol, page 217](#)
- [Additional References, page 220](#)
- [Feature Information for Rate Based Satellite Control Protocol, page 221](#)

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

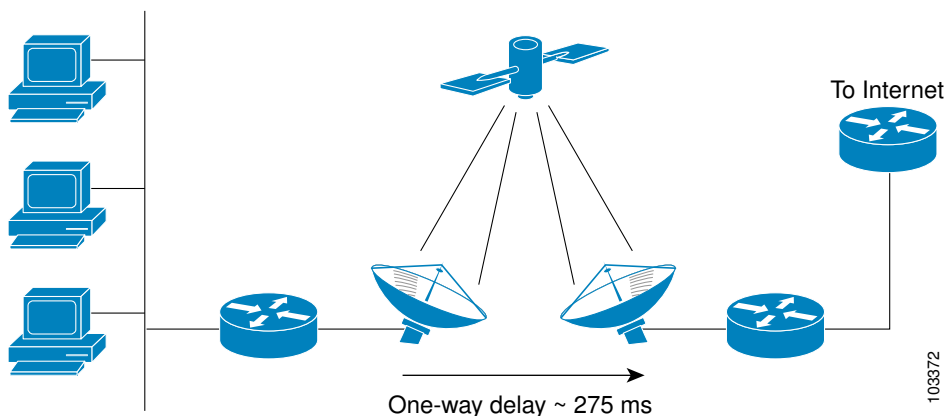
- [IP over Satellite Links, page 210](#)

- [Performance Enhancing Proxy over Satellite Links](#), page 210
- [RBSCP over Satellite Links](#), page 211

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 16 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.

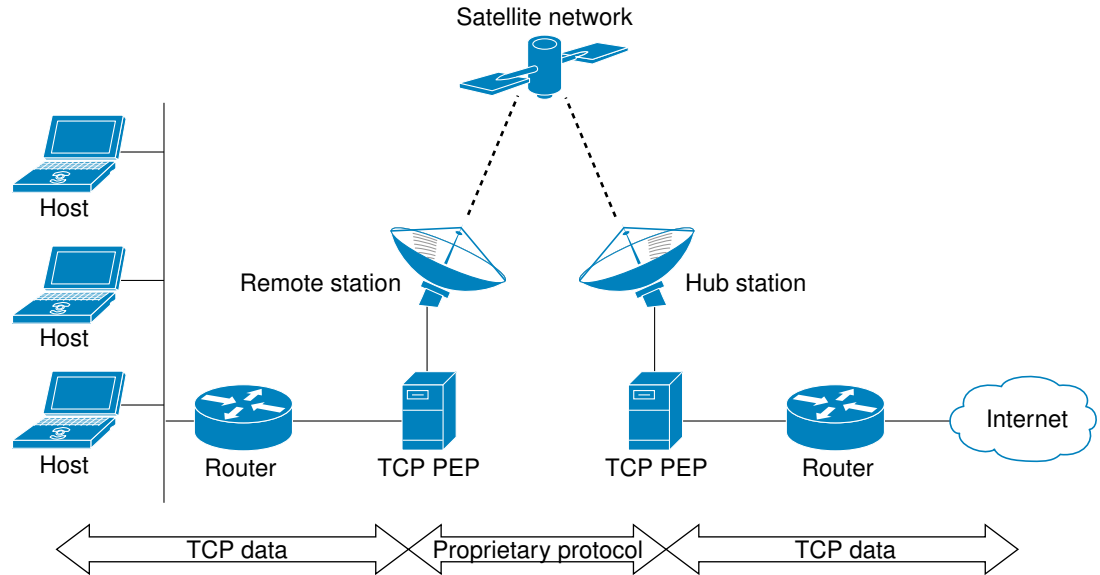
Although available satellite link bandwidths are increasing, the long RTT and high error rates experienced by IP protocols over satellite links are producing a high bandwidth-delay product (BDP).

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 17 *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.

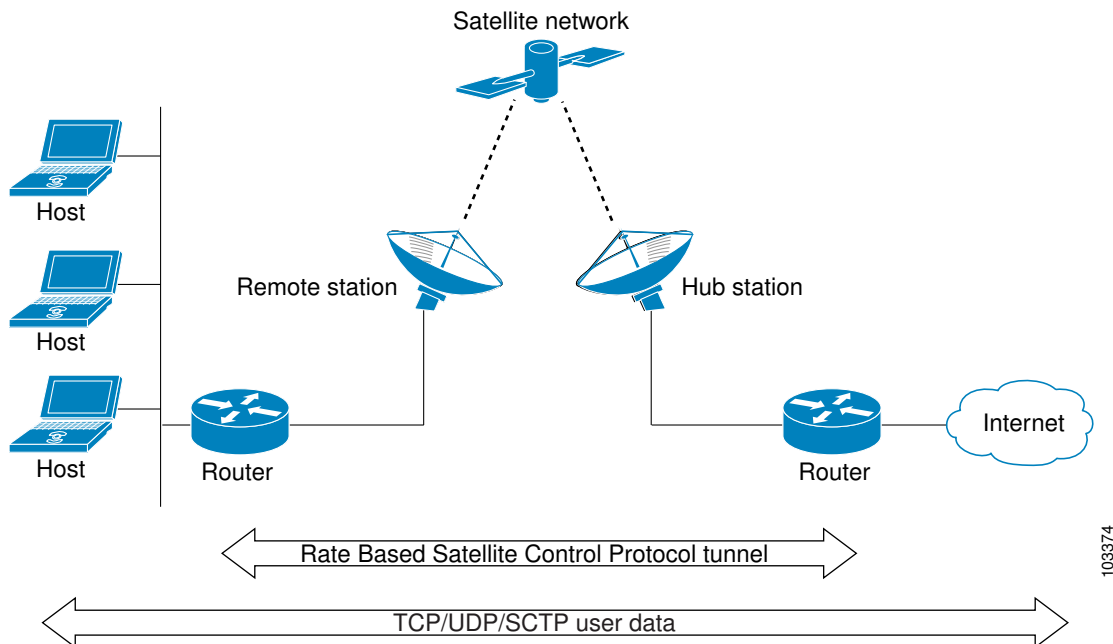
RBSCP over Satellite Links

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 18 **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, page 213](#)
- [Verifying that the RBSCP Tunnel Is Active, page 216](#)
- [Verifying the RBSCP Traffic, page 217](#)

Configuring the RBSCP Tunnel

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

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** | **ipop**[**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	<p>enable</p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>interface <i>type number</i></p> <p>Example:</p> <pre>Router(config)# interface tunnel 0</pre>	<p>Specifies the interface type and number and enters interface configuration mode.</p>
Step 4	<p>ip unnumbered <i>interface-type interface-number</i></p> <p>Example:</p> <pre>Router(config-if)# ip unnumbered Ethernet 1</pre>	<p>Enables IP processing on an interface without assigning an explicit IP address.</p> <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	<p>tunnel source {<i>ip-address</i> <i>interface-type interface-number</i>}</p> <p>Example:</p> <pre>Router(config-if)# tunnel source Ethernet 1</pre>	<p>Configures the tunnel source.</p> <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	<p>tunnel destination {<i>hostname</i> <i>ip-address</i>}</p> <p>Example:</p> <pre>Router(config-if)# tunnel destination 172.17.2.1</pre>	<p>Configures the tunnel destination.</p> <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.

Command or Action	Purpose
<p>Step 7 <code>tunnel bandwidth {receive transmit} bandwidth</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel bandwidth transmit 1000</pre>	<p>Specifies the tunnel bandwidth to be used to transmit packets.</p> <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>
<p>Step 8 <code>tunnel mode {aurp cayman dvmp eon gre gre multipoint ipip[decapsulate-any] iptalk mpls nos rbscp}</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel mode rbscp</pre>	<p>Specifies the protocol to be used in the tunnel.</p> <ul style="list-style-type: none"> Use the rbscp keyword to specify that RBSCP will be used as the tunnel protocol.
<p>Step 9 <code>tunnel rbscp ack_split split-size</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel rbscp ack_split 6</pre>	<p>(Optional) Enables TCP acknowledgement (ACK) splitting with RBSCP tunnels.</p> <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.
<p>Step 10 <code>tunnel rbscp delay</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel rbscp delay</pre>	<p>(Optional) Enables RBSCP tunnel delay.</p> <ul style="list-style-type: none"> Use this command only when the RTT is greater than 700 milliseconds.
<p>Step 11 <code>tunnel rbscp report</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel rbscp report</pre>	<p>(Optional) Reports dropped RBSCP packets to SCTP.</p> <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.
<p>Step 12 <code>tunnel rbscp window_stuff step-size</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel rbscp window_stuff 3</pre>	<p>(Optional) Enables TCP window stuffing by increasing the value of the TCP window scale for RBSCP tunnels.</p> <ul style="list-style-type: none"> Use the <i>step-size</i> argument to specify the step increment number.

Command or Action	Purpose
Step 13 <code>exit</code> Example: <code>Router(config-if)# exit</code>	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, page 216](#)

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, page 213](#) task is active.

SUMMARY STEPS

- `enable`
- `show rbscp [all| state| statistics] [tunnel tunnel-number]`
- `exit`

DETAILED STEPS

Command or Action	Purpose
Step 1 <code>enable</code> Example: <code>Router> enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.
Step 2 <code>show rbscp [all state statistics] [tunnel tunnel-number]</code> Example: <code>Router# show rbscp state tunnel 1</code>	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.

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

Configuration Examples for Rate Based Satellite Control Protocol

- [Configuring the RBSCP Tunnel Example, page 218](#)
- [Configuring Routing for the RBSCP Tunnel Example, page 218](#)

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 rbsp window_stuff 1
tunnel rbsp ack_split 4
!
interface FastEthernet0/0
ip address 10.13.0.1 255.255.255.0
ip policy route-map rbsp-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 rbsp-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 rbsp-pbr permit 10
match ip address rbsp-acl
set interface Tunnel1

```

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 Tunnel1
ip unnumbered FastEthernet0/1
tunnel source FastEthernet0/1
tunnel destination 10.12.0.1
tunnel mode rbsp
tunnel ttl 5
tunnel bandwidth transmit 30000
tunnel rbsp window_stuff 1
tunnel rbsp ack_split 4
!
interface FastEthernet0/0
description Local LAN
ip address 10.15.0.1 255.255.255.0
ip policy route-map rbsp-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 rbsp-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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 13 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	<p>This feature was introduced.</p> <p>The following commands were introduced or modified:</p> <p>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.</p>

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



Configuring Virtual Interfaces

Virtual interfaces are software-based interfaces that you create in the memory of the networking device using Cisco IOS commands. Virtual interfaces do not have a hardware component such as the RJ-45 female port on a 100BASE-T Fast Ethernet network interface card. This module describes the four common types of virtual, or logical, interfaces that can be configured using Cisco IOS software:

- Loopback interfaces
- Null interfaces
- Subinterfaces
- Tunnel interfaces

- [Finding Feature Information, page 223](#)
- [Prerequisites for Configuring Virtual Interfaces, page 223](#)
- [Information About Configuring Virtual Interfaces, page 224](#)
- [How to Configure Virtual Interfaces, page 228](#)
- [Configuration Examples for Virtual Interfaces, page 234](#)
- [Where to Go Next, page 234](#)
- [Additional References, page 235](#)
- [Feature Information for Configuring Virtual Interfaces, page 236](#)

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.

Prerequisites for Configuring Virtual Interfaces

Before virtual interfaces can be used in your network, you must have some physical (hardware) interfaces configured and be able to communicate between the networking devices on which you want to use virtual interfaces.

Information About Configuring Virtual Interfaces

- [Virtual Interfaces](#), page 224
- [Benefits of Virtual Interfaces](#), page 224
- [Loopback Interfaces](#), page 225
- [Loopback Interfaces Versus Loopback Mode](#), page 226
- [Null Interfaces](#), page 226
- [Subinterfaces](#), page 227
- [Tunnel Interfaces](#), page 227
- [Virtual Multipoint Interface](#), page 228

Virtual Interfaces

Virtual interfaces are network interfaces that are not associated with a physical interface. Physical interfaces have some form of physical element—for example, an RJ-45 male connector on an Ethernet cable. Virtual interfaces exist only in software; there are no physical elements. You identify an individual virtual interface using a numerical ID after the virtual interface name. For example: loopback 0, tunnel 1, and fastethernet 0/0.1. The ID is unique per virtual interface type to make the entire name string unique; for example, both a loopback 0 interface and a null 0 interface can exist, but two loopback 0 interfaces cannot exist in a single networking device.

Cisco IOS software supports four types of virtual interfaces;

- Loopback
- Null
- Subinterface
- Tunnel

Benefits of Virtual Interfaces

A loopback interface can provide a stable interface on which you can assign a Layer 3 address such as an IP or IPX address. This address can be configured as the source address when the networking device needs to send data for protocols such as NetFlow or Cisco Discovery Protocol (CDP) to another device in your network and you want the receiving device to always see the same source IP address from the networking device. This is an issue in networks with multiple equal-cost paths because under normal circumstances the packets that are generated by a networking device use the IP address from the outbound interface as the source address for the packets and because in a network with two or more equal-cost paths from the networking device to the receiving host each packet might use a different outbound interface.

A null interface provides an alternative method of filtering without the overhead involved with using access lists. For example, instead of creating an outbound access list that prevents traffic to a destination network from being transmitted out an interface, you can configure a static route for the destination network that points to the null interface.

Subinterfaces were invented as a method of virtually subdividing a physical interface into two or more interfaces so that the IP routing protocols would see the network connection to each remote networking device as a separate physical interface even though the subinterfaces share a common physical interface. One of the first uses of subinterfaces was to resolve the problem with split horizon on Frame Relay WANs.

The following are several situations in which tunneling (encapsulating traffic in another protocol) is useful:

- To enable multiprotocol local networks over a single-protocol backbone

- To provide workarounds for networks that use protocols that have limited hop counts; for example, RIP version 1, AppleTalk
- To connect discontinuous subnetworks
- To allow virtual private networks across WANs

Loopback Interfaces

You can specify a software-only interface called a loopback interface to emulate a physical interface. Loopback interfaces are supported on all platforms. A loopback interface is a virtual interface on a Cisco router that remains up (active) after you issue the **no shutdown** command until you disable it with the **shutdown** command. Unlike subinterfaces, loopback interfaces are independent of the state of any physical interface.

The loopback interface can be considered stable because once you enable it, it will remain up until you shut it down. This makes loopback interfaces ideal for assigning Layer 3 addresses such as IP addresses when you want a single address as a reference that is independent of the status of any physical interfaces in the networking device. A good example of this is using the IP address of a loopback interface as the IP address for the domain name system (DNS) host address for the networking device. Before loopback interfaces were available, network administrators had to configure a DNS host entry for every interface on a router that had an IP address assigned to it because they could never be certain which interface IP address might be available at any given time for managing the router. In the following sample interface configuration and DNS entries for Router A, you can see that there is a DNS entry for each interface.

Router A Interface Configuration Before Loopback

```
Ethernet0 10.10.10.1 255.255.255.224
Ethernet1 10.10.11.1 255.255.255.224
Ethernet2 10.10.12.1 255.255.255.224
Ethernet3 10.10.13.1 255.255.255.224
Ethernet4 10.10.14.1 255.255.255.224
Ethernet5 10.10.15.1 255.255.255.224
```

Router A DNS Entries Before Loopback

```
RouterA    IN  A  10.10.10.1
           IN  A  10.10.11.1
           IN  A  10.10.12.1
           IN  A  10.10.13.1
           IN  A  10.10.14.1
           IN  A  10.10.15.1
```

Interfaces on networking devices can fail, and they can also be taken out of service for maintenance. If any of the interfaces in Router A fails or is taken out of service, another networking device will not be able to access that interface. When you configure a networking device with a loopback interface and assign it an IP address that is advertised throughout the network, the networking device will be reachable by using this IP address as long as the networking device has at least one network interface capable of sending and receiving IP traffic. In the sample interface configuration and DNS entries for Router A after a loopback interface is configured, you can see that there is now only one DNS entry that can be used to reach the router over any of its physical interfaces.

Router A Interface Configuration After Loopback

```
Loopback 172.16.78.1 255.255.255.224
Ethernet0 10.10.10.1 255.255.255.224
Ethernet1 10.10.11.1 255.255.255.224
Ethernet2 10.10.12.1 255.255.255.224
```

```
Ethernet3 10.10.13.1 255.255.255.224  
Ethernet4 10.10.14.1 255.255.255.224  
Ethernet5 10.10.15.1 255.255.255.224
```

Router A DNS Entries After Loopback

```
RouterA IN A 172.16.78.1
```

The configured IP address of the loopback interface--172.16.78.1--can be used as the source address for packets generated by the router and forwarded to networking management applications and routing protocols. Unless this loopback interface is explicitly shut down, it is always reachable.

You can use the loopback interface as the termination address for open shortest path first (OSPF) or border gateway protocol (BGP) sessions. A loopback interface can also be used to establish a Telnet session from the console port of the device to its auxiliary port when all other interfaces are down. In applications where other routers or access servers attempt to reach this loopback interface, you should configure a routing protocol to distribute the subnet assigned to the loopback address.

IP packets routed to the loopback interface are rerouted back to the router or access server and processed locally. IP packets routed out the loopback interface but not destined to the loopback interface are dropped. Under these two conditions, the loopback interface can behave like a null interface.

Loopback Interfaces Versus Loopback Mode

Loopback interfaces provide a stable source interface to ensure that the IP address assigned to the interface is always reachable as long as the IP routing protocols continue to advertise the subnet assigned to the loopback interface. Loopback mode, however, is used to test and diagnose issues with WAN (serial) links such as bit loss or data corruption. The idea is to configure a loop to return the data packets that were received by the interface back out the same interface to the device that originated the traffic. Loopback mode is used to troubleshoot problems by checking that the data packets are returned in the same condition in which they were sent. Errors in the data packets indicate a problem with the WAN infrastructure. Many types of serial interfaces have their own form of loopback command syntax that is entered in interface or controller configuration mode.

For more details about loopback mode, see the "Configuring Serial Interfaces" chapter.

Null Interfaces

The null interface is a virtual network interface that is similar to the loopback interface. Whereas traffic to the loopback interface is directed to the router itself, traffic sent to the null interface is discarded. This interface is always up and can never forward or receive traffic; encapsulation always fails. The null interface functions similarly to the null devices available on most operating systems.

Null interfaces are used as a low-overhead method of discarding unnecessary network traffic. For example, if you do not want your network users to be able to reach certain IP subnets, you can create static IP routes for the subnets that point to the null interface of a networking device. Using the static IP routes takes less CPU time for the networking device than using IP access lists. The static-route configuration is also easier to configure than IP access lists because it is done in global configuration mode instead of in interface configuration mode.

The null interface may not be configured with an address. Traffic can be sent to this interface only by configuring a static route where the next hop is the null interface--represented by Null 0. One example of configuring the next hop to be the null interface is to create a route to an aggregate network that can then be announced through the BGP, or to ensure that traffic to a particular range of addresses is not propagated through the router, perhaps for security purposes.

The router always has a single null interface. By default, a packet sent to the null interface causes the router to respond by sending an Internet Control Message Protocol (ICMP) unreachable message to the source IP

address of the packet. You can configure the router either to send these responses or to drop the packets silently.

Subinterfaces

Subinterfaces are associated with physical interfaces. Subinterfaces are enabled when the physical interface with which they are associated is enabled, and subinterfaces are disabled when the physical interface is shut down.



Note

Subinterfaces can be enabled and shut down independently of the physical port with which they are associated. However, you cannot enable a subinterface of a physical interface that has been shut down.

Subinterfaces are created by subdividing the physical interface into two or more virtual interfaces on which you can assign unique Layer 3 network addresses such as IP subnets. One of the first uses of subinterfaces was to resolve the problem with split horizon on Frame Relay WANs. Split horizon is a behavior associated with IP routing protocols such as Routing Information Protocol (RIP) and OSPF in which IP subnets are not advertised back out the same physical interface that they were learned over. Split horizon was implemented to prevent routing loops in IP networks. A routing loop can be created when the networking devices at both ends of a network connection advertise the same IP routes to each other. Split horizon was an issue for Frame Relay multipoint network interfaces--interfaces that connect to two or more remote networking devices over a single physical interface--because the default behavior of many networking devices was to implement split horizon, which means that the networking device did not advertise the IP routes that were learned over an interface back out the interface to other devices that were also reachable via the same physical interface. Subinterfaces were invented as a method of virtually subdividing a physical interface into two or more interfaces so that the IP routing protocols would see the network connection to each remote networking device as a separate physical interface even though the subinterfaces share a common physical interface. Although TCP/IP now disables split horizon limitations by default, protocols such as AppleTalk and IPX are still constrained by split horizon.

Subinterfaces are identified by a prefix that consists of the hardware interface descriptor (IDB) followed by a period and then by a number that is unique for that prefix. The full subinterface number must be unique to the networking device. For example, the first subinterface for Ethernet interface 0/0 might be named Ethernet 0/0.1 where .1 indicates the subinterface.

Tunnel Interfaces

Tunneling provides a way to encapsulate arbitrary packets inside a transport protocol. Tunnels are implemented as a virtual interface to provide a simple interface for configuration. The tunnel interface is not tied to specific "passenger" or "transport" protocols, but, rather, it is an architecture that is designed to provide the services necessary to implement any standard point-to-point encapsulation scheme.

There are several ways to implement tunnel interfaces depending on the connectivity that you need to provide. One common use for tunnels is to carry data traffic for a network protocol such as IPX over devices in your network that do not support IPX. For instance, if your network uses IPX in sites at the edge of your network but not in the core of your network, you can connect the IPX sites at the network edges by tunneling IPX in IP over the core of the network.

For more details about the various types of tunneling techniques available using Cisco IOS software, see the "Implementing Tunnels" module.

Virtual Multipoint Interface

Used in router-to-radio communications, the Virtual Multipoint Interface (VMI) interface provides services that map outgoing packets to the appropriate Point-to-Point Protocol over Ethernet (PPPoE) sessions based on the next-hop forwarding address for that packet. The VMI interface also provides a broadcast service that emulates a set of point-to-point connections as a point-to-multipoint interface with broadcast ability. When a packet with a multicast address is forwarded through the VMI interface, VMI replicates the packet and unicasts it to each of its neighbors.

Directional radios are frequently used in applications that require greater bandwidth, increased power-to-transmission range, or reduced probability of detection. These radios operate in a point-to-point mode, and generally have no broadcast capability. However, the routing processes in Cisco's Mobile Adhoc Networks (MANET) solution operate most efficiently when viewing the network link as point-to-multipoint, with broadcast capability. For the router, modeling the MANET as a collection of point-to-point nodes would have a dramatic impact on the size of its internal database.

The VMI within the router aggregates all of the per-neighbor PPPoE sessions from the Radio Ethernet connection. The VMI maps the sessions to appear to Layer 3 routing protocols and applications as a single point-to-multipoint, multiaccess, broadcast-capable network. However, the VMI preserves the integrity of the PPPoE sessions on the radio side, so that each point-to-point connection can have its own quality of service (QoS) queue.

The VMI also relays the link quality metric and neighbor up/down signaling from the radio to the routing protocols. Currently, VMI signals are used by enhanced interior gateway routing protocol (EIGRP) (for IPv4 and IPv6 neighbors) and OSPFv3 (for IPv6 neighbors).

For more details about the VMI interface, see the " Mobile Adhoc Networks for Router-to-Radio Communications" module in the Cisco IOS IP Mobility Configuration Guide.

How to Configure Virtual Interfaces

- [Configuring a Loopback Interface, page 228](#)
- [Configuring a Null Interface, page 230](#)
- [Configuring a Subinterface, page 232](#)

Configuring a Loopback Interface

This task explains how to configure a loopback interface. A loopback interface can be considered stable because once you enable it, it will remain up until you shut it down. This makes loopback interfaces ideal for assigning Layer 3 addresses such as IP addresses to when you want to have a single address to use as a reference that is independent of the status of any of the physical interfaces in the networking device.

The IP address for the loopback interface must be unique and not in use by another interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface loopback** *number*
4. **ip address** *ip-address mask* [**secondary**]
5. **end**
6. **show interfaces loopback** *number*
7. **exit**

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 enable</p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
<p>Step 2 configure terminal</p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
<p>Step 3 interface loopback <i>number</i></p> <p>Example:</p> <pre>Router(config)# interface loopback 0</pre>	<p>Specifies a loopback interface and enters interface configuration mode.</p> <ul style="list-style-type: none"> • Use the <i>number</i> argument to specify the number of the loopback interface that you want to create or configure. <p>Note There is no limit on the number of loopback interfaces that you can create.</p>
<p>Step 4 ip address <i>ip-address mask</i> [secondary]</p> <p>Example:</p> <pre>Router(config-if)# ip address 209.165.200.225 255.255.255.224</pre>	<p>Specifies an IP address for the loopback interface and enables IP processing on the interface.</p> <ul style="list-style-type: none"> • Use the <i>ip-address</i> and <i>mask</i> arguments to specify the subnet for the loopback address.
<p>Step 5 end</p> <p>Example:</p> <pre>Router(config-if)# end</pre>	<p>Exits interface configuration mode and returns to privileged EXEC mode.</p>

Command or Action	Purpose
<p>Step 6 <code>show interfaces loopback <i>number</i></code></p> <p>Example:</p> <pre>Router# show interfaces loopback 0</pre>	<p>(Optional) Displays information about loopback interfaces.</p> <ul style="list-style-type: none"> Use the <i>number</i> argument to display information about one particular loopback interface. <p>Note Only the syntax applicable to this task is used in this example.</p>
<p>Step 7 <code>exit</code></p> <p>Example:</p> <pre>Router# exit</pre>	<p>Exits privileged EXEC mode.</p>

Examples

The following is sample output from the `show interfaces loopback` command:

```
Router# show interfaces loopback 0
Loopback0 is up, line protocol is up
  Hardware is Loopback
  Internet address is 10.20.1.2/24
  MTU 1514 bytes, BW 8000000 Kbit, DLY 5000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation LOOPBACK, loopback not set
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 no buffer
    Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    0 packets output, 0 bytes, 0 underruns
    0 output errors, 0 collisions, 0 interface resets
    0 output buffer failures, 0 output buffers swapped out
```

Configuring a Null Interface

This task explains how to configure a null interface. Null interfaces provide an alternative method to access control lists for filtering traffic. All unwanted traffic can be directed to the null interface; the null interface cannot receive or forward traffic, or allow its traffic to be encapsulated.

The only interface configuration command that you can specify for the null interface is the `no ip unreachable` command.

- [ICMP Unreachable Messages from Null Interfaces, page 230](#)

ICMP Unreachable Messages from Null Interfaces

By default, a packet sent to the null interface causes the router to respond by sending an Internet Control Message Protocol (ICMP) unreachable message to the source IP address of the packet. You can configure the router either to send these responses or to drop the packets silently.

To disable the sending of ICMP unreachable messages in response to packets sent to the null interface, use the `no ip unreachable` command in interface configuration mode. To reenble the sending of ICMP

unreachable messages in response to packets sent to the null interface, use the **ip unreachable** command in interface configuration mode.



Note Only one null interface can be configured on each networking device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface null** *number*
4. **no ip unreachable**s
5. **end**
6. **show interfaces null** [*number*] [**accounting**]

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 configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3 interface null <i>number</i> Example: Router(config)# interface null 0	Specifies a null interface and number, and enters interface configuration mode. <ul style="list-style-type: none"> • The number argument is always 0.
Step 4 no ip unreachable s Example: Router(config-if)# no ip unreachable	Prevents the generation of ICMP unreachable messages on an interface. <ul style="list-style-type: none"> • This command affects all types of ICMP unreachable messages.

Command or Action	Purpose
Step 5 <code>end</code> Example: Router(config-if)# <code>end</code>	Exits interface configuration mode and returns to privileged EXEC mode.
Step 6 <code>show interfaces null [number] [accounting]</code> Example: Router# <code>show interfaces null 0</code>	(Optional) Displays information about null interfaces. <ul style="list-style-type: none"> For null interfaces, the <i>number</i> argument is always 0. Note Only the syntax applicable to this task is used in this example.

Examples

The following is sample output from the **show interfaces null** command:

```
Router# show interfaces null
Null0 is up, line protocol is up
  Hardware is Unknown
  MTU 1500 bytes, BW 10000000 Kbit, DLY 0 usec,
    reliability 0/255, txload 0/255, rxload 0/255
  Encapsulation ARPA, loopback not set
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 no buffer
  Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    0 packets output, 0 bytes, 0 underruns
    0 output errors, 0 collisions, 0 interface resets
    0 output buffer failures, 0 output buffers swapped out
```

Configuring a Subinterface

This task explains how to configure a subinterface. Subinterfaces can be enabled and shut down independently of the physical port with which they are associated. However, you cannot enable a subinterface of a physical interface that has been shut down.

The IP address for the interface must be unique and not in use by another interface.

SUMMARY STEPS

- enable**
- configure terminal**
- interface** *type number.subinterface-number*
- ip address** *ip-address mask [secondary]*
- end**
- show interfaces** *type number.subinterface-number*
- exit**

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 <code>enable</code></p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
<p>Step 2 <code>configure terminal</code></p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
<p>Step 3 <code>interface type number.subinterface-number</code></p> <p>Example:</p> <pre>Router(config)# interface GigabitEthernet 2/3.5</pre>	<p>Specifies the interface type, interface number, and subinterface number and enters interface configuration mode.</p>
<p>Step 4 <code>ip address ip-address mask [secondary]</code></p> <p>Example:</p> <pre>Router(config-if)# ip address 209.165.200.225 255.255.255.224</pre>	<p>Specifies an IP address for the interface and enables IP processing on the interface.</p>
<p>Step 5 <code>end</code></p> <p>Example:</p> <pre>Router(config-if)# end</pre>	<p>Exits interface configuration mode and returns to privileged EXEC mode.</p>
<p>Step 6 <code>show interfaces type number.subinterface-number</code></p> <p>Example:</p> <pre>Router# show interfaces GigabitEthernet 2/3.5</pre>	<p>(Optional) Displays information about the interfaces.</p>
<p>Step 7 <code>exit</code></p> <p>Example:</p> <pre>Router# exit</pre>	<p>Exits privileged EXEC mode.</p>

Examples

The following is sample output from the **show interfaces** command:

```
Router# show interfaces GigabitEthernet 2/3.5
GigabitEthernet2/3.5432 is down, line protocol is down (notconnect)
  Hardware is c7600 1Gb 802.3, address is 001b.0de6.c100 (bia 001b.0de6.c100)
  Description: *sample*
  Internet address is 10.11.12.13/24
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation 802.1Q Virtual LAN, Vlan ID 2339.
  ARP type: ARPA, ARP Timeout 04:00:00
  Keepalive set (10 sec)
  Last clearing of "show interface" counters never
```

Configuration Examples for Virtual Interfaces

- [Configuring a Loopback Interface Example, page 234](#)
- [Configuring a Null Interface Example, page 234](#)
- [Configuring a Subinterface Example, page 234](#)

Configuring a Loopback Interface Example

The following example shows the configuration sequence of a loopback interface, loopback 0:

```
interface loopback 0
ip address 209.165.200.225 255.255.255.0
end
```

Configuring a Null Interface Example

The following example shows the configuration sequence of a null interface and how to drop the ICMP unreachable messages. All packets sent to the null interface are dropped and in this example, the ICMP messages usually sent in response to packets being sent to the null interface are dropped.

```
interface null 0
no ip unreachable
end
```

Configuring a Subinterface Example

The following example shows the configuration sequence of a subinterface:

```
interface GigabitEthernet 2/3.5
description *sample*
encapsulation dot1Q 2339
ip address 209.165.200.225 255.255.255.224
end
```

Where to Go Next

- If you want to implement tunnels in your network, see the "Implementing Tunnels" module.

- If you want to implement other types of interfaces such as LAN or serial in your network, see the "Configuring LAN Interfaces" or the "Configuring Serial Interfaces" chapters.

Additional References

The following sections provide references related to virtual interfaces.

Related Documents

Related Topic	Document Title
Interface commands: complete command syntax, command mode, defaults, command history, usage guidelines, and examples	<i>Cisco IOS Interface and Hardware Component Command Reference</i>
Implementing Tunnels	Implementing Tunnels module
Configuring LAN interfaces	Configuring LAN Interfaces module
Configuring Serial interfaces	Configuring Serial Interfaces module
Configuration example showing how to use loopback interfaces with BGP	Sample Configuration for iBGP and eBGP With or Without a Loopback Address

Standards

Standard	Title
No new or modified standards are supported, and support for existing standards has not been modified.	--

MIBs

MIB	MIBs Link
No new or modified MIBs are supported, and support for existing MIBs has not been modified.	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

RFC	Title
No new or modified RFCs are supported, and support for existing RFCs has not been modified.	--

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/techsupport</p>

Feature Information for Configuring Virtual Interfaces

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 14 *Feature Information for Configuring Virtual Interfaces*

Feature Name	Releases	Feature Information
Virtual Multipoint Interface	12.4(15)T	VMI interface provides services that map outgoing packets to the appropriate Point-to-Point Protocol over Ethernet (PPPoE) sessions based on the next-hop forwarding address for that packet.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams,

and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



Implementing Tunnels

This module describes the various types of tunneling techniques available using Cisco IOS software. Configuration details and examples are provided for the tunnel types that use physical or virtual interfaces. Many tunneling techniques are implemented using technology-specific commands, and links are provided to the appropriate technology modules.

Tunneling provides a way to encapsulate arbitrary packets inside a transport protocol. Tunnels are implemented as a virtual interface to provide a simple interface for configuration. The tunnel interface is not tied to specific "passenger" or "transport" protocols, but rather is an architecture to provide the services necessary to implement any standard point-to-point encapsulation scheme.



Note

Cisco ASR 1000 Series Aggregation Services Routers support VPN routing and forwarding (VRF)-aware generic routing encapsulation (GRE) tunnel keepalive features.

- [Finding Feature Information](#), page 239
- [Prerequisites for Implementing Tunnels](#), page 239
- [Restrictions for Implementing Tunnels](#), page 240
- [Information About Implementing Tunnels](#), page 241
- [How to Implement Tunnels](#), page 259
- [Configuration Examples for Implementing Tunnels](#), page 288
- [Where to Go Next](#), page 301
- [Additional References](#), page 302
- [Feature Information for Implementing Tunnels](#), page 304

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.

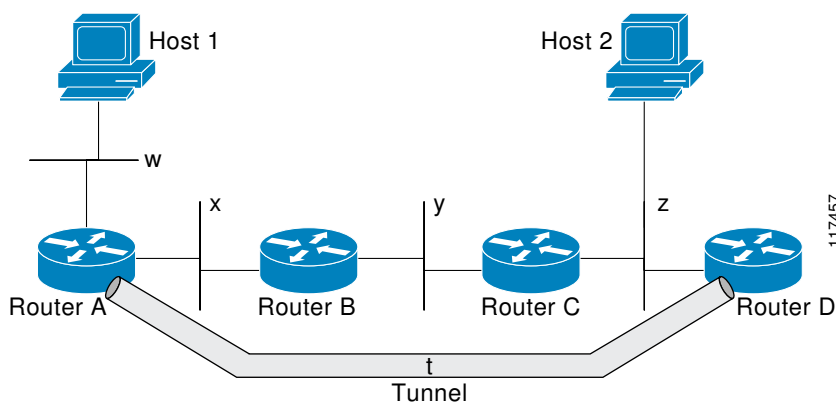
Prerequisites for Implementing Tunnels

This module assumes that you are running Cisco IOS Release 12.2 software or a later release.

Restrictions for Implementing Tunnels

- In early versions of Cisco IOS software, only processor switching was supported. Fast switching of GRE tunnels was introduced in Cisco IOS Release 11.1. Cisco Express Forwarding (CEF) switching is also now commonly used by the IPv6 and other tunneling protocols.
- It is important to allow the tunnel protocol through a firewall and to allow it to pass access control list (ACL) checking.
- Multiple point-to-point tunnels can saturate the physical link with routing information if the bandwidth is not configured correctly on the tunnel interface.
- A tunnel looks like a single hop, and routing protocols may prefer a tunnel over a multihop physical path. This can be deceptive because the tunnel, although it may look like a single hop, may traverse a slower path than a multihop link. A tunnel is as robust and fast, or as unreliable and slow, as the links that it actually traverses. Routing protocols that make their decisions on the sole basis of hop count will often prefer a tunnel over a set of physical links. A tunnel might appear to be a one-hop, point-to-point link and have the lowest-cost path, but may actually cost more in terms of latency than an alternative physical topology. For example, in the topology shown in the figure below, packets from Host 1 will appear to travel across networks w, t, and z to get to Host 2 instead of taking the path w, x, y, and z because the tunnel hop count appears shorter. In fact, the packets going through the tunnel will still be traveling across Router A, B, and C, but they must also travel to Router D before coming back to Router C.

Figure 19 Tunnel Precautions: Hop Counts



- If routing is not carefully configured, the tunnel may have a recursive routing problem. When the best path to the "tunnel destination" is via the tunnel itself, recursive routing causes the tunnel interface to flap. To avoid recursive routing problems, keep the control-plane routing separate from the tunnel routing using the following methods:
 - Use a different autonomous system number or tag.
 - Use a different routing protocol.
 - Use static routes to override the first hop (but watch for routing loops).

When you have recursive routing to the tunnel destination, the following error is displayed:

```
%TUN-RECURDOWN Interface Tunnel 0
temporarily disabled due to recursive routing
```

**Note**

You cannot configure an IP tunnel or a GRE tunnel on Cisco 7600 series routers which has an MPLS Traffic Engineering (TE) tunnel as the egress path, because the configuration results in forwarding loops.

Information About Implementing Tunnels

- [Tunneling Versus Encapsulation](#), page 241
- [Definition of Tunneling Types by OSI Layer](#), page 242
- [Benefits of Tunneling](#), page 246
- [Tunnel ToS](#), page 247
- [Mobile IP Tunneling](#), page 247
- [Generic Routing Encapsulation](#), page 248
- [GRE Tunnel IP Source and Destination VRF Membership](#), page 248
- [EoMPLS over GRE](#), page 248
- [Multipoint GRE Tunneling](#), page 250
- [GRE CLNS Tunnel Support for IPv4 and IPv6 Packets](#), page 250
- [GRE IPv4 Tunnel Support for IPv6 Traffic](#), page 250
- [Overlay Tunnels for IPv6](#), page 251
- [IPv6 Manually Configured Tunnels](#), page 253
- [Automatic 6to4 Tunnels](#), page 253
- [Automatic IPv4-Compatible IPv6 Tunnels](#), page 254
- [ISATAP Tunnels](#), page 254
- [Rate-Based Satellite Control Protocol Tunnels](#), page 255
- [Path MTU Discovery](#), page 258
- [QoS Options for Tunnels](#), page 258

Tunneling Versus Encapsulation

To understand how tunnels work, it is important to distinguish between the concepts of encapsulation and tunneling. Encapsulation is the process of adding headers to data at each layer of a particular protocol stack. The Open Systems Interconnection (OSI) reference model describes the functions of a network as seven layers stacked on top of each other. When data has to be sent from one host (a PC for example) on a network to another host, the process of encapsulation is used to add a header in front of the data at each layer of the protocol stack in descending order. The header must contain a data field that indicates the type of data encapsulated at the layer immediately above the current layer. As the packet ascends the protocol stack on the receiving side of the network, each encapsulation header is removed in the reverse order.

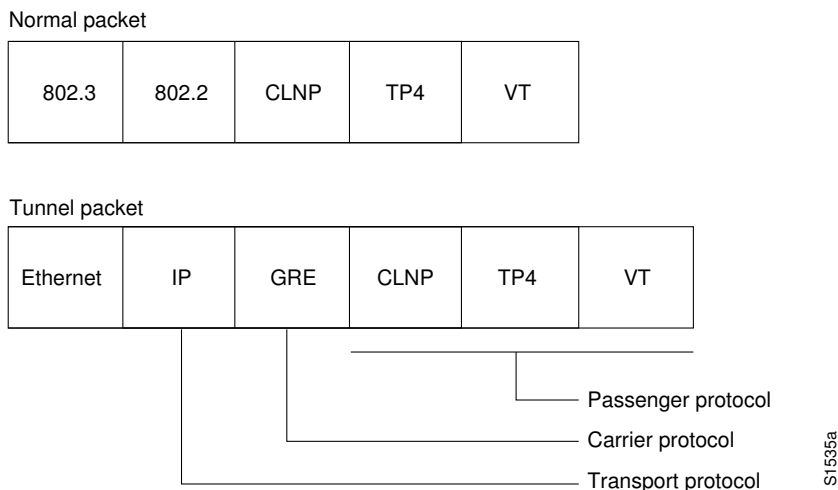
Tunneling encapsulates data packets from one protocol inside a different protocol and transports the data packets unchanged across a foreign network. Unlike encapsulation, tunneling allows a lower-layer protocol, or same-layer protocol, to be carried through the tunnel. A tunnel interface is a virtual (or logical) interface. For more details on other types of virtual interfaces, see the "Configuring Virtual Interfaces" module. Although many different types of tunnels have been created to solve different network problems, tunneling consists of three main components:

- **Passenger protocol**--The protocol that you are encapsulating. Examples of passenger protocols are AppleTalk, connectionless network service (CLNS), IP, and IPX.

- Carrier protocol--The protocol that does the encapsulating. Examples of carrier protocols are GRE, IP-in-IP, Layer 2 Tunneling Protocol (L2TP), multiprotocol label switching (MPLS), STUN, and DLSw +.
- Transport protocol--The protocol used to carry the encapsulated protocol. The main transport protocol is IP.

The figure below illustrates IP tunneling terminology and concepts.

Figure 20 IP Tunneling Terminology and Concepts



To understand the process of tunneling, consider connecting two AppleTalk networks with a non-AppleTalk backbone, such as IP. The relatively high bandwidth consumed by the broadcasting of Routing Table Maintenance Protocol (RTMP) data packets can severely hamper the backbone’s network performance. This problem can be solved by tunneling AppleTalk through a foreign protocol, such as IP. Tunneling encapsulates an AppleTalk packet inside the foreign protocol packet (AppleTalk inside GRE inside IP), which is then sent across the backbone to a destination router. The destination router then removes the encapsulation from the AppleTalk packet and routes the packet.

Definition of Tunneling Types by OSI Layer

Tunnels are used by many different technologies to solve different network challenges, and the resulting variety of tunnel types makes it difficult to determine which tunneling technique to use. The different carrier protocols can be grouped according to the OSI layer model. The table below shows the different carrier protocols grouped by OSI layer. Below the table, each carrier protocol is defined, and if the tunnel configuration is not covered within this module, a link to the appropriate module is included.

Table 15 Carrier Protocol by OSI Layer

Layer	Carrier Protocol
2	<ul style="list-style-type: none"> • PPPoA--Point-to-Point Protocol (PPP) over ATM • PPPoE--PPP over Ethernet • UDLR--Unidirectional link routing

Layer	Carrier Protocol
3	<ul style="list-style-type: none"> • BSTUN--Block Serial Tunneling • CLNS--Connectionless Network Service (CLNS) • GRE--Generic routing encapsulation • IP-in-IP--Internet Protocol encapsulated within IP • IPsec--IP Security • IPv6--IP version 6 • L2F--Layer 2 Forwarding • L2TP--Layer 2 Tunneling Protocol • MPLS--Multiprotocol Label Switching • PPTP--Point-to-Point Tunneling Protocol • STUN--Serial Tunneling
4	<ul style="list-style-type: none"> • DLSw+--Data-link switching plus • RBSCP--Rate-Based Satellite Control Protocol • SSL--Secure Socket Layer

BSTUN

A Block Serial Tunnel (BSTUN) enables support for devices using the Bisync data-link protocol. This protocol enables enterprises to transport Bisync traffic over the same network that supports their Systems Network Architecture (SNA) and multiprotocol traffic, eliminating the need for separate Bisync facilities.

For more details about configuring BSTUN, see the "Configuring Serial Tunnel and Block Serial Tunnel" module in the Cisco IOS Bridging and IBM Networking Configuration Guide.

CLNS

The ISO Connectionless Network Service (CLNS) protocol is a standard for the network layer of the OSI model. IP traffic can be transported over CLNS; for instance, on the data communications channel (DCC) of a SONET ring. An IP over CLNS tunnel (CTunnel) is a virtual interface that enhances interactions with CLNS networks, allowing IP packets to be tunneled through the Connectionless Network Protocol (CLNP) to preserve TCP/IP services. CLNS can also be used as a transport protocol with GRE as a carrier protocol (GRE/CLNS), carrying both IPv4 and IPv6 packets.

DLSw+

Data-link switching plus (DLSw+) is Cisco's implementation of the DLSw standard for Systems Network Architecture (SNA) and NetBIOS devices, and it supports several additional features and enhancements. DLSw+ is a means of transporting SNA and NetBIOS traffic over a campus or WAN. The end systems can attach to the network over Token Ring, Ethernet, Synchronous Data Link Control (SDLC), Qualified Logical Link Control (QLLC), or Fiber Distributed Data Interface (FDDI). DLSw+ switches between diverse media and locally terminates the data links, keeping acknowledgments, keepalives, and polling off the WAN.

For more details about configuring DLSw+, see the "Configuring Data-Link Switching Plus" module in the Cisco IOS Bridging and IBM Networking Configuration Guide .

GRE

Generic routing encapsulation (GRE) is defined in RFC 2784. GRE is a carrier protocol that can be used with a variety of underlying transport protocols, and GRE can carry a variety of passenger protocols. RFC 2784 also covers the use of GRE with IPv4 as the transport protocol and the passenger protocol. Cisco IOS software supports GRE as the carrier protocol with many combinations of passenger and transport protocols.

For more details about GRE, see the [Generic Routing Encapsulation, page 248](#).

IP-in-IP

IP-in-IP is a Layer 3 tunneling protocol--defined in RFC 2003--that alters the normal routing of an IP packet by encapsulating it within another IP header. The encapsulating header specifies the address of a router that would not ordinarily be selected as a next-hop router on the basis of the real destination address of the packet. The intermediate node decapsulates the packet, which is then routed to the destination as usual.

IPsec

In simple terms, IP Security (IPsec) provides secure tunnels between two peers, such as two routers. You define which packets are considered sensitive and should be sent through these secure tunnels, and you define the parameters that should be used to protect these packets by specifying characteristics of these tunnels. IPsec peers set up a secure tunnel and encrypt the packets that traverse the tunnel to the remote peer.

IPsec also works with the GRE and IP-in-IP, L2F, L2TP, and DLSw+ tunneling protocols; however, multipoint tunnels are not supported. Other Layer 3 tunneling protocols may not be supported for use with IPsec.

For more details about configuring IPsec, see the "Configuring Security for VPNs with IPsec" module in the Cisco IOS Security Configuration Guide.

IPv6

IP version 6 (IPv6) is a new version of the Internet Protocol based on and designed as the successor to IP version 4. IPv6 adds a much larger address space--128 bits--and improvements such as a simplified main header and extension headers. IPv6 is described initially in RFC 2460, *Internet Protocol, Version 6 (IPv6)*. The use of IPv6 as a carrier protocol is described in RFC 2473, *Generic Packet Tunneling in IPv6 Specification*.

L2F

Layer 2 Forwarding (L2F) tunneling is used in virtual private dialup networks (VPDNs). A VPDN allows separate and autonomous protocol domains to share common access infrastructure including modems, access servers, and ISDN routers by the tunneling of link-level (Layer 2) frames. Typical L2F tunneling use includes Internet service providers (ISPs) or other access service creating virtual tunnels to link to remote customer sites or remote users with corporate intranet or extranet networks.

L2TP

Layer 2 Tunneling Protocol (L2TP) is an open standard created by the Internet Engineering Task Force (IETF) that uses the best features of L2F and Point-to-Point Tunneling Protocol (PPTP). L2TP is designed to secure the transmission of IP packets across uncontrolled and untrusted network domains, and it is an important component of Virtual Private Networks (VPNs). VPNs extend remote access to users over a shared infrastructure while maintaining the same security and management policies as a private network.

For more details about configuring L2TP, see the Cisco IOS Dial Technologies Configuration Guide.

MPLS

Multiprotocol Label Switching (MPLS) is a high-performance packet forwarding technology that integrates the performance and traffic management capabilities of data-link-layer (Layer 2) switching with the scalability, flexibility, and performance of network-layer (Layer 3) routing. The MPLS architecture has been designed to allow data to be transferred over any combination of Layer 2 technologies, to support all Layer 3 protocols, and to scale. Using CEF, MPLS can efficiently enable the delivery of IP services over an ATM switched network. MPLS is an integration of Layer 2 and Layer 3 technologies. By making traditional Layer 2 features available to Layer 3, MPLS enables traffic engineering.

For more details about how MPLS traffic engineering uses tunnels, see the Cisco IOS Multiprotocol Label Switching Configuration Guide.

PPPoA

PPP over ATM (PPPoA) is mainly implemented as part of Asymmetric Digital Subscriber Line (ADSL). It relies on RFC 1483, operating in either Logical Link Control-Subnetwork Access Protocol (LLC-SNAP) or VC-Mux mode. A customer premises equipment (CPE) device encapsulates the PPP session based on this RFC for transport across the ADSL loop and the digital subscriber line access multiplexer (DSLAM).

PPPoE

RFC 2516 defines PPP over Ethernet (PPPoE) as providing the ability to connect a network of hosts over a simple bridging access device to a remote access concentrator or aggregation concentrator. As customers deploy ADSL, they must support PPP-style authentication and authorization over a large installed base of legacy bridging customer premises equipment (CPE). Using a form of tunneling encapsulation, PPPoE allows each host to use its own PPP stack, thus presenting the user with a familiar user interface. Access control, billing, and type of service (ToS) can be done on a per-user, rather than a per-site, basis.

For more details about configuring PPPoE, see the Cisco IOS Broadband Access Aggregation and DSL Configuration Guide.

PPTP

Point-to-Point Tunneling Protocol (PPTP) is a network protocol that enables the secure transfer of data from a remote client enterprise server by creating a VPN across TCP/IP data networks. PPTP supports on-demand, multiprotocol virtual private networking over public networks such as the Internet.

RBSCP

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.

SSL Tunnels

Secure Socket Layer (SSL) is designed to make use of TCP sessions to provide a reliable end-to-end secure service. The main role of SSL is to provide security for web traffic. Security includes confidentiality, message integrity, and authentication. SSL achieves these elements of security through the use of cryptography, digital signatures, and certificates. SSL protects confidential information through the use of cryptography. Sensitive data is encrypted across public networks to achieve a level of confidentiality.

SSL is implemented using the Cisco Application and Content Networking System (ACNS). For more details about configuring SSL, see the latest [Cisco ACNS Software Deployment and Configuration Guide](#).

STUN

Cisco's Serial Tunneling (STUN) implementation allows Synchronous Data Link Control (SDLC) protocol devices and High-Level Data Link Control (HDLC) devices to connect to one another through a multiprotocol internetwork rather than through a direct serial link. STUN encapsulates SDLC frames in either the TCP/IP or the HDLC protocol. STUN provides a straight passthrough of all SDLC traffic (including control frames, such as Receiver Ready) end-to-end between Systems Network Architecture (SNA) devices.

For more details about configuring STUN, see the "Configuring Serial Tunnel and Block Serial Tunnel" module in the [Cisco IOS Bridging and IBM Networking Configuration Guide](#).

UDLR Tunnels

Unidirectional link routing (UDLR) provides mechanisms for a router to emulate a bidirectional link to enable the routing of unicast and multicast packets over a physical unidirectional interface, such as a broadcast satellite link. However, there must be a back channel or other path between the routers that share a physical unidirectional link (UDL). A UDLR tunnel is a mechanism for unicast and multicast traffic; Internet Group Management Protocol (IGMP) UDLR is a related technology for multicast traffic.

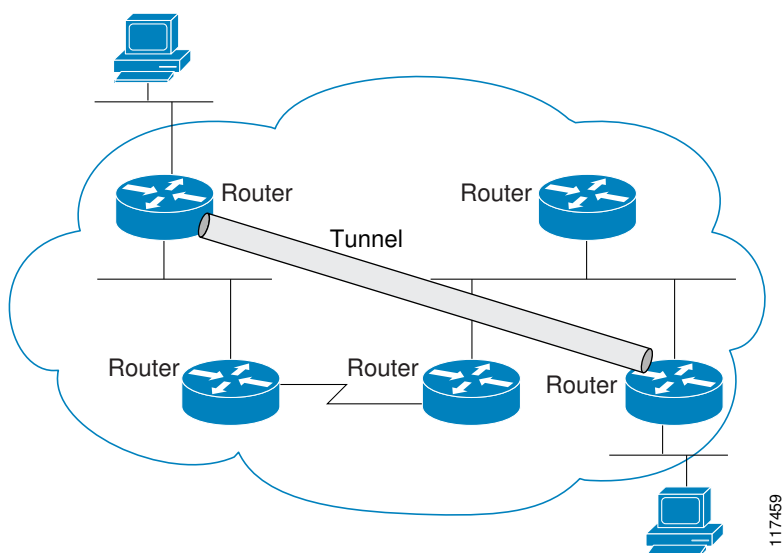
For more details, see Cisco IOS IP Multicast Configuration Guide.

Benefits of Tunneling

The following are several situations in which tunneling (encapsulating traffic in another protocol) is useful:

- To enable multiprotocol local networks over a single-protocol backbone.
- To provide workarounds for networks that use protocols that have limited hop counts; for example, RIP version 1, AppleTalk (see the figure below).
- To connect discontinuous subnetworks.
- To allow virtual private networks across WANs.

Figure 21 Providing Workarounds for Networks with Limited Hop Counts



If the path between two computers has more than 15 hops, the computers cannot communicate with each other, but it is possible to hide some of the hops inside the network using a tunnel.

Tunnel ToS

Tunnel ToS allows you to tunnel your network traffic and group all your packets in the same specific ToS byte value. The ToS byte values and Time-to-Live (TTL) hop-count value can be set in the encapsulating IP header of tunnel packets for an IP tunnel interface on a router. The Tunnel ToS feature is supported for CEF, fast switching, and process switching.

The ToS and TTL byte values are defined in RFC 791. RFC 2474 and RFC 2780 obsolete the use of the ToS byte as defined in RFC 791. RFC 791 specifies that bits 6 and 7 of the ToS byte (the first two least significant bits) are reserved for future use and should be set to 0. The Tunnel ToS feature does not conform to this standard and allows you to set the whole ToS byte value, including bits 6 and 7, and to decide to which RFC standard the ToS byte of your packets should conform.

Mobile IP Tunneling

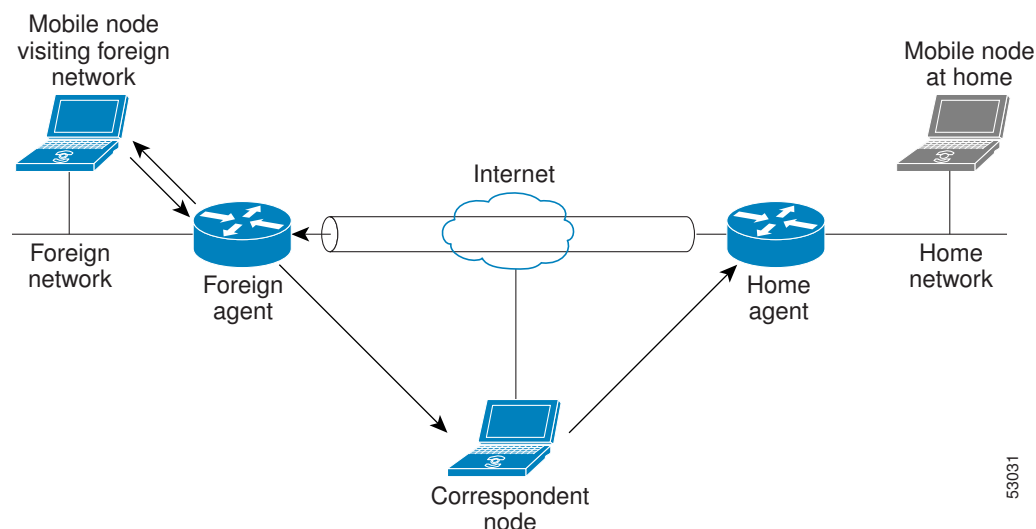
New devices and business practices, such as PDAs and the next-generation of data-ready cellular phones and services, are driving interest in the ability of a user to roam while maintaining network connectivity. The requirement for data connectivity solutions for this group of users is very different than it is for the fixed dialup user or the stationary wired LAN user. Solutions need to accommodate the challenge of movement during a data session or conversation.

Mobile IP is a tunneling-based solution that takes advantage of the Cisco-created generic routing encapsulation (GRE) tunneling technology and simpler IP-in-IP tunneling protocol.

Mobile IP comprises the following three components, as shown in the figure below:

- Mobile node (MN)
- Home agent (HA)
- Foreign agent (FA)

Figure 22 Mobile IP Components and Use of Tunneling



An MN is a node, for example, a PDA, a laptop computer, or a data-ready cellular phone, that can change its point of attachment from one network or subnet to another. This node can maintain ongoing communications while using only its home IP address. In the figure above, the current location of the MN--a laptop computer--is shown in bold.

An HA is a router on the home network of the MN that maintains an association between the home IP address of the MN and its *care-of address*, which is the current location of the MN on a foreign or visited network. The HA redirects packets by tunneling them to the MN while it is away from home.

An FA is a router on a foreign network that assists the MN in informing its HA of its current care-of address. The FA detunnels packets that were tunneled by the HA and delivers them to the MN. The FA also acts as the default router for packets generated by the MN while it is connected to the foreign network.

The traffic destined for the MN is forwarded in a triangular manner. When a device on the Internet, called a correspondent node (CN), sends a packet to the MN, the packet is routed to the home network of the MN, the HA redirects the packet by tunneling to the care-of address (current location) of the MN on the foreign network, as shown in the figure above. The FA receives the packet from the HA and forwards it locally to the MN. However, packets sent by the MN are routed directly to the CN.

For more details about configuring Mobile IP, see the Cisco IOS IP Mobility Configuration Guide.

Generic Routing Encapsulation

Generic routing encapsulation (GRE) is defined in RFC 2784. GRE is a carrier protocol that can be used with a variety of underlying transport protocols and that can carry a variety of passenger protocols. RFC 2784 also covers the use of GRE with IPv4 as the transport protocol and the passenger protocol. Cisco IOS software supports GRE as the carrier protocol with many combinations of passenger and transport protocols such as:

The following descriptions of GRE tunnels are included:

GRE Tunnel IP Source and Destination VRF Membership

GRE Tunnel IP Source and Destination VRF Membership allows you to configure the source and destination of a tunnel to belong to any VRF tables. A VRF table stores routing data for each VPN. The VRF table defines the VPN membership of a customer site attached to the network access server (NAS). Each VRF table comprises an IP routing table, a derived CEF table, and guidelines and routing protocol parameters that control the information that is included in the routing table.

Previously, Generic Routing Encapsulation (GRE) IP tunnels required the IP tunnel destination to be in the global routing table. The implementation of this feature allows you to configure a tunnel source and destination to belong to any VRF. As with existing GRE tunnels, the tunnel becomes disabled if no route to the tunnel destination is defined.

EoMPLS over GRE

Ethernet over multiprotocol label switching (EoMPLS) is a tunneling mechanism that allows you to tunnel Layer 2 traffic through a Layer 3 MPLS network. EoMPLS is also known as Layer 2 tunneling.

EoMPLS effectively facilitates the Layer 2 extension over long distances. EoMPLS over GRE helps to create the GRE tunnel as hardware-based switched, and with high performance that encapsulates EoMPLS frames within the GRE tunnel. The GRE connection is established between the two core routers, and then the MPLS LSP is tunneled over.

GRE encapsulation is used to define a packet that has some additional header information added to it prior to being forwarded. De-encapsulation is the process of removing the additional header information when the packet reaches the destination tunnel endpoint.

When a packet is forwarded through a GRE tunnel, two new headers are appended at the front of the packet and hence the context of the new payload changes. After encapsulation, what was originally the data payload and separate IP header is now known as the GRE payload. A GRE header is added to the packet to

provide information on the protocol type and also a recalculated checksum. Also, a new IP header is added to the front of the GRE header. This IP header contains the destination IP address of the tunnel.

The GRE header is appended to the packet (IP, L2VPN, L3VPN, etc.) before entering the tunnel. All routers along the path that receive the encapsulated packet will use the new IP header to determine where to send the packet in an effort for it to reach the tunnel endpoint.

In the IP forwarding case on reaching the tunnel destination endpoint, the new IP header and GRE header are removed from the packet and the original IP header is then used to forward the packet to its final destination.

In the EoMPLS over GRE cases, the new IP header and GRE header will be removed from the packet at the tunnel destination and the MPLS (VC or VPN) label will be used to forward the packets to the appropriate L2 attachment circuit or L3 VRF.

The following scenarios describe the L2VPN and L3VPN over GRE deployment on provider edge (PE) or provider (P) routers:

PE to PE GRE Tunnels

In the PE to PE GRE tunnels scenario, a customer does not generally transition any part of the core to MPLS but prefers to offer EoMPLS and basic MPLS VPN services. Hence, GRE tunneling of the MPLS labeled traffic is done between PEs. This is the most common scenario seen in various customers networks.

P to P GRE Tunnels

The P to P GRE tunnels scenario is one where MPLS has been enabled between PE and P routers, but the network core may have non-MPLS-aware routers or IP encryption boxes. In this scenario, GRE tunneling of the MPLS labeled packets is done between P routers.

PE to P GRE Tunnels

The PE to P GRE tunnels scenario demonstrates a network where the P to P nodes are MPLS-aware, while GRE tunneling is done between a PE to P non MPLS network segment.

The following features are required for the deployment of scenarios described above:

GRE Specific:

- Tunnel endpoints can be loopbacks or physical interfaces.
- Configurable tunnel keepalive timer parameters per end point, and syslog message must be generated when the keepalive timer expires.
- BFD support for tunnel failures and for IGP using tunnels.
- IGP loadsharing across GRE tunnels.
- IGP redundancy across GRE tunnels.
- Fragmentation across GRE tunnels.
- Ability to pass jumbo frames.
- Support for all IGP control plane traffic.
- Support for IP TOS preservation across tunnel.
- Tunnel should be independent of endpoint physical interface types such as POS, Gig, TenGig, and ATM.
- Support for up to 100 GRE tunnels.

EoMPLS Specific:

- Port mode EoMPLS.
- VLAN mode EoMPLS.
- Pseudowire redundancy.
- AToM sequencing.
- Tunnel selection--ability to map a specific pseudowire or pw-class to a GRE tunnel.
- IGP loadsharing and redundancy. See below for more information.
- Support for up to 200 EoMPLS VCs.

MPLS-VPN Specific:

- Support for PE Role with IPv4 VRFs
- Support for all PE-CE protocols
- Load sharing through multiple tunnels and equal-cost IGP paths with a single tunnel
- Support for redundancy via unequal-cost IGP paths with a single tunnel
- Support for the IP Precedence value being copied onto the EXP bits field of the MPLS label, and then onto the Precedence bits on the outer IPv4 ToS field of the GRE packet.

For a sample configuration sequence of EoMPLS over GRE, see [Example Configuring EoMPLS over GRE](#), page 300. For more details about EoMPLS over GRE, see [Deploying and Configuring MPLS Virtual Private Networks In IP Tunnel Environments](#).

Multipoint GRE Tunneling

Enhanced multipoint GRE (mGRE) tunneling technology provides a Layer 3 (L3) transport mechanism for use in IP networks. This same dynamic Layer 3 tunneling transport can be used within IP networks to transport VPN traffic across service provider and enterprise networks, as well as to provide interoperability for packet transport between IP and MPLS VPNs. This feature provides support for RFC 2547, which defines the outsourcing of IP-backbone services for enterprise networks.

Multipoint tunnels use the Next Hop Resolution Protocol (NHRP) in the same way that a Frame Relay multipoint interface uses information obtained by the reverse ARP mechanism to learn the Layer 3 addresses of the remote data-link connection identifiers (DLCIs).

In Cisco IOS Release 12.2(8)T and later releases, CEF-switching over mGRE tunnels was introduced. Previously, only process switching was available for mGRE tunnels. CEF-switching over mGRE tunnels enables CEF switching of IP traffic to and from multipoint GRE tunnels. Tunnel traffic can be forwarded to a prefix through a tunnel destination when both the prefix and the tunnel destination are specified by the application.

GRE CLNS Tunnel Support for IPv4 and IPv6 Packets

GRE tunneling of IPv4 and IPv6 packets through CLNS networks enables Cisco CLNS tunnels (CTunnels) to interoperate with networking equipment from other vendors. This feature provides compliance with RFC 3147.

The optional GRE services defined in header fields, such as checksums, keys, and sequencing, are not supported. Any packet that is received and requests such services will be dropped.

GRE IPv4 Tunnel Support for IPv6 Traffic

IPv6 traffic can be carried over IPv4 generic routing encapsulation (GRE) tunnels using the standard GRE tunneling technique that is designed to provide the services necessary to implement any standard point-to-

point encapsulation scheme. As in IPv6 manually configured tunnels, GRE tunnels are links between two points, with a separate tunnel for each link. The tunnels are not tied to a specific passenger or transport protocol, but in this case, IPv6 is the passenger protocol, GRE is the carrier protocol, and IPv4 is the transport protocol.

The primary use of GRE tunnels is for stable connections that require regular secure communication between two edge routers or between an edge router and an end system. The edge routers and the end systems must be dual-stack implementations.

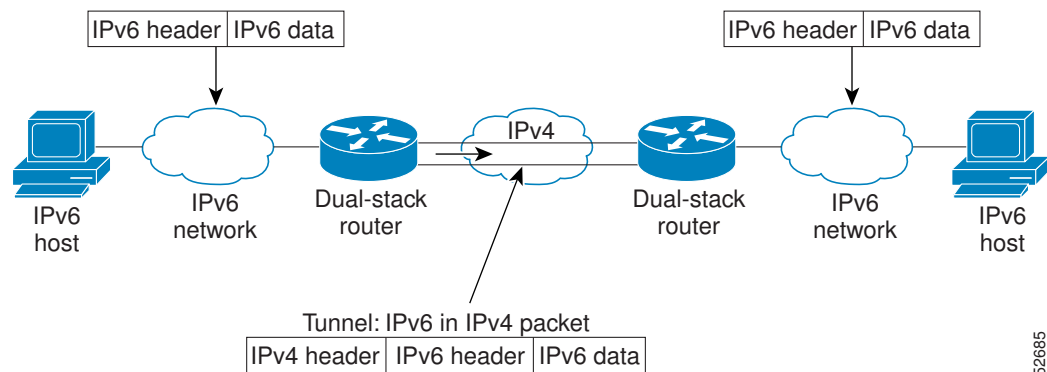
GRE has a protocol field that identifies the passenger protocol. GRE tunnels allow IS-IS or IPv6 to be specified as a passenger protocol, allowing both IS-IS and IPv6 traffic to run over the same tunnel. If GRE did not have a protocol field, it would be impossible to distinguish whether the tunnel was carrying IS-IS or IPv6 packets. The GRE protocol field is why it is desirable that you tunnel IS-IS and IPv6 inside GRE.

Overlay Tunnels for IPv6

Overlay tunneling encapsulates IPv6 packets in IPv4 packets for delivery across an IPv4 infrastructure (a core network or the Internet). (See the figure below.) By using overlay tunnels, you can communicate with isolated IPv6 networks without upgrading the IPv4 infrastructure between them. Overlay tunnels can be configured between border routers or between a border router and a host; however, both tunnel endpoints must support both the IPv4 and IPv6 protocol stacks. Cisco IOS IPv6 currently supports the following types of overlay tunneling mechanisms:

- Manual
- Generic routing encapsulation (GRE)
- IPv4-compatible
- 6to4
- Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)

Figure 23 *Overlay Tunnels*



52685



Note

Overlay tunnels reduce the maximum transmission unit (MTU) of an interface by 20 octets (assuming that the basic IPv4 packet header does not contain optional fields). A network that uses overlay tunnels is difficult to troubleshoot. Therefore, overlay tunnels that connect isolated IPv6 networks should not be considered as a final IPv6 network architecture. The use of overlay tunnels should be considered as a transition technique toward a network that supports both the IPv4 and IPv6 protocol stacks or just the IPv6 protocol stack.

Use the table below to help you determine which type of tunnel you want to configure to carry IPv6 packets over an IPv4 network.

Table 16 *Suggested Usage of Tunnel Types to Carry IPv6 Packets over an IPv4 Network*

Tunneling Type	Suggested Usage	Usage Notes
Manual	Simple point-to-point tunnels that can be used within a site or between sites.	Can carry IPv6 packets only.
GRE/IPv4	Simple point-to-point tunnels that can be used within a site or between sites.	Can carry IPv6, CLNS, and many other types of packets.
Compatible	Point-to-multipoint tunnels.	Uses the ::/96 prefix. Currently, we do not recommend using this tunnel type.
6to4	Point-to-multipoint tunnels that can be used to connect isolated IPv6 sites.	Sites use addresses from the 2002::/16 prefix.
ISATAP	Point-to-multipoint tunnels that can be used to connect systems within a site.	Sites can use any IPv6 unicast addresses.

Individual tunnel types are discussed in more detail in the following concepts, and we recommend that you review and understand the information on the specific tunnel type that you want to implement. When you are familiar with the type of tunnel you need, the table below provides a quick summary of the tunnel configuration parameters that you may find useful.

Table 17 *Overlay Tunnel Configuration Parameters by Tunneling Type*

Overlay Tunneling Type	Overlay Tunnel Configuration Parameter	Tunnel Destination	Interface Prefix/Address	
Manual	ipv6ip	An IPv4 address or a reference to an interface on which IPv4 is configured.	An IPv4 address.	An IPv6 address.
GRE/IPv4	gre ip		An IPv4 address.	An IPv6 address.
Compatible	ipv6ip auto-tunnel		Not required. These are all point-to-multipoint tunneling types. The IPv4 destination address is calculated, on a per-packet basis,	Not required. The interface address is generated as :: <i>tunnel-source</i> /96.

Overlay Tunneling Type	Overlay Tunnel Configuration Parameter		
6to4	ipv6ip 6to4	from the IPv6 destination.	An IPv6 address. The prefix must embed the tunnel source IPv4 address.
ISATAP	ipv6ip isatap		An IPv6 prefix in modified eui-64 format. The IPv6 address is generated from the prefix and the tunnel source IPv4 address.

IPv6 Manually Configured Tunnels

A manually configured tunnel is equivalent to a permanent link between two IPv6 domains over an IPv4 backbone. The primary use is for stable connections that require regular secure communication between two edge routers or between an end system and an edge router, or for connection to remote IPv6 networks.

An IPv6 address is manually configured on a tunnel interface, and manually configured IPv4 addresses are assigned to the tunnel source and the tunnel destination. The host or router at each end of a configured tunnel must support both the IPv4 and IPv6 protocol stacks. Manually configured tunnels can be configured between border routers or between a border router and a host. CEF switching can be used for IPv6 manually configured tunnels, or CEF switching can be disabled if process switching is needed.

Automatic 6to4 Tunnels

An automatic 6to4 tunnel allows isolated IPv6 domains to be connected over an IPv4 network to remote IPv6 networks. The key difference between automatic 6to4 tunnels and manually configured tunnels is that the tunnel is not point-to-point; it is point-to-multipoint. In automatic 6to4 tunnels, routers are not configured in pairs because they treat the IPv4 infrastructure as a virtual nonbroadcast multiaccess (NBMA) link. The IPv4 address embedded in the IPv6 address is used to find the other end of the automatic tunnel.

An automatic 6to4 tunnel may be configured on a border router in an isolated IPv6 network, which creates a tunnel on a per-packet basis to a border router in another IPv6 network over an IPv4 infrastructure. The tunnel destination is determined by the IPv4 address of the border router extracted from the IPv6 address that starts with the prefix 2002::/16, where the format is 2002:*border-router-IPv4-address*::/48. Following the embedded IPv4 address are 16 bits that can be used to number networks within the site. The border router at each end of a 6to4 tunnel must support both the IPv4 and IPv6 protocol stacks. 6to4 tunnels are configured between border routers or between a border router and a host.

The simplest deployment scenario for 6to4 tunnels is to interconnect multiple IPv6 sites, each of which has at least one connection to a shared IPv4 network. This IPv4 network could be the global Internet or a corporate backbone. The key requirement is that each site have a globally unique IPv4 address; the Cisco IOS software uses this address to construct a globally unique 6to4/48 IPv6 prefix. As with other tunnel

mechanisms, appropriate entries in a Domain Name System (DNS) that map between hostnames and IP addresses for both IPv4 and IPv6 allow the applications to choose the required address.

Automatic IPv4-Compatible IPv6 Tunnels

Automatic IPv4-compatible tunnels use IPv4-compatible IPv6 addresses. IPv4-compatible IPv6 addresses are IPv6 unicast addresses that have zeros in the high-order 96 bits of the address and an IPv4 address in the low-order 32 bits. They can be written as 0:0:0:0:0:A.B.C.D or ::A.B.C.D, where "A.B.C.D" represents the embedded IPv4 address.

The tunnel destination is automatically determined by the IPv4 address in the low-order 32 bits of IPv4-compatible IPv6 addresses. The host or router at each end of an IPv4-compatible tunnel must support both the IPv4 and IPv6 protocol stacks. IPv4-compatible tunnels can be configured between border routers or between a border router and a host. Using IPv4-compatible tunnels is an easy method to create tunnels for IPv6 over IPv4, but the technique does not scale for large networks.



Note

IPv4-compatible tunnels were initially supported for IPv6, but are currently being deprecated. Cisco now recommends that you use a different IPv6 tunneling technique named ISATAP tunnels.

ISATAP Tunnels

The Intra-Site Automatic Tunnel Addressing Protocol (ISATAP) is an automatic overlay tunneling mechanism that uses the underlying IPv4 network as a nonbroadcast multiaccess (NBMA) link layer for IPv6. ISATAP is designed for transporting IPv6 packets within a site where a native IPv6 infrastructure is not yet available; for example, when sparse IPv6 hosts are deployed for testing. ISATAP tunnels allow individual IPv4/IPv6 dual-stack hosts within a site to communicate with other such hosts on the same virtual link, basically creating an IPv6 network using the IPv4 infrastructure.

The ISATAP router provides standard router advertisement network configuration support for the ISATAP site. This feature allows clients to automatically configure themselves as they would do if they were connected to an Ethernet. It can also be configured to provide connectivity out of the site. ISATAP uses a well-defined IPv6 address format composed of any unicast IPv6 prefix (/64), which can be link-local or global (including 6to4 prefixes), enabling IPv6 routing locally or on the Internet. The IPv4 address is encoded in the last 32 bits of the IPv6 address, enabling automatic IPv6-in-IPv4 tunneling.

While the ISATAP tunneling mechanism is similar to other automatic tunneling mechanisms, such as IPv6 6to4 tunneling, ISATAP is designed for transporting IPv6 packets within a site, *not between* sites.

ISATAP uses unicast addresses that include a 64-bit IPv6 prefix and a 64-bit interface identifier. The interface identifier is created in modified EUI-64 format in which the first 32 bits contain the value 000:5EFE to indicate that the address is an IPv6 ISATAP address. The table below shows the layout of an ISATAP address.

Table 18 **ISATAP address example**

64 Bits	32 Bits	32 Bits
Link local or global IPv6 unicast prefix	0000:5EFE	IPv4 address of the ISATAP link

As shown in the table above, an ISATAP address consists of an IPv6 prefix and the ISATAP interface identifier. This interface identifier includes the IPv4 address of the underlying IPv4 link. The following

example shows what an actual ISATAP address would look like if the prefix is 2001:0DB8:1234:5678::/64 and the embedded IPv4 address is 10.173.129.8. In the ISATAP address, the IPv4 address is expressed in hexadecimal as 0AAD:8108.

Example

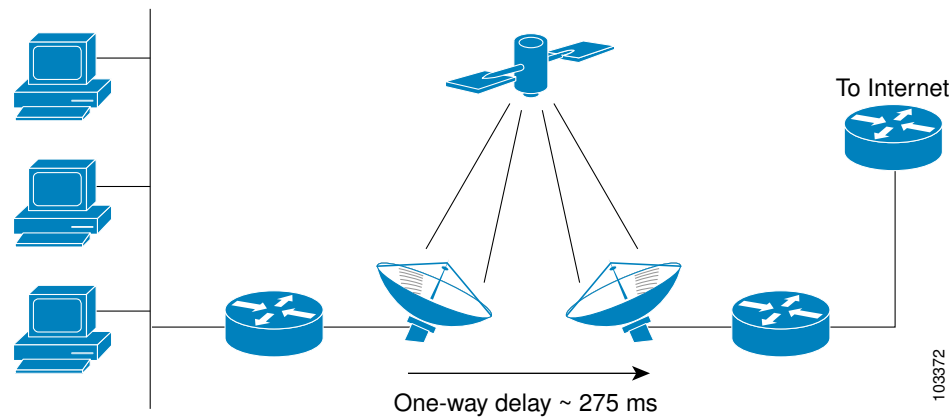
```
2001:0DB8:1234:5678:0000:5EFE:0AAD:8108
```

Rate-Based Satellite Control Protocol Tunnels

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.

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 24 Typical Satellite Link



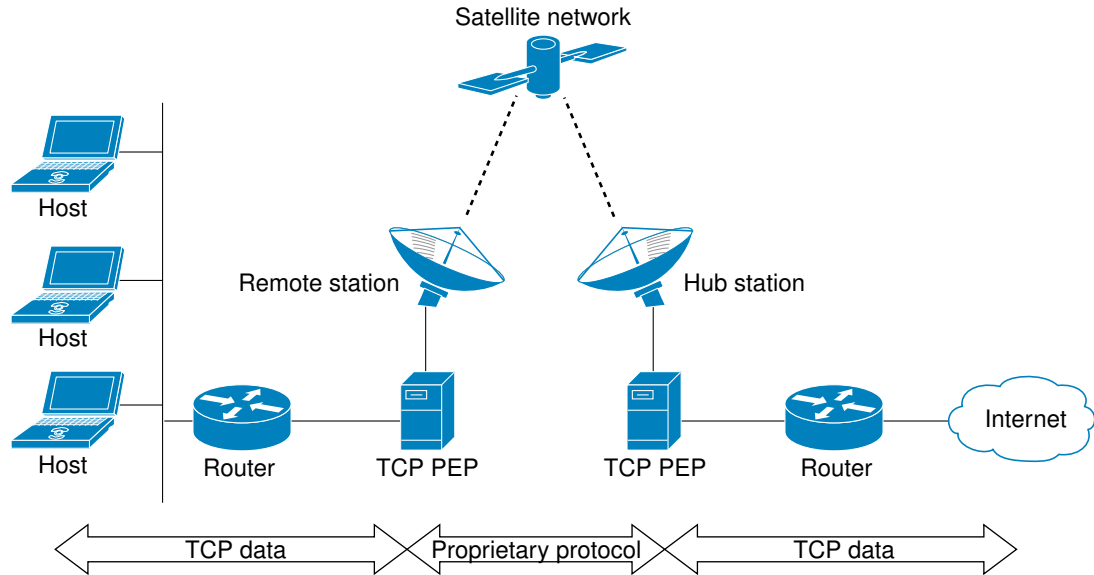
Long RTT keeps TCP in a slow start mode, which increases the time before the satellite link bandwidth is fully used. TCP and Stream Control Transmission Protocol (SCTP) interpret packet loss events as congestion in the network and start to perform congestion recovery procedures, which reduce the traffic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

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 often 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 25 *Disruptive TCP PEP Solution*



103373

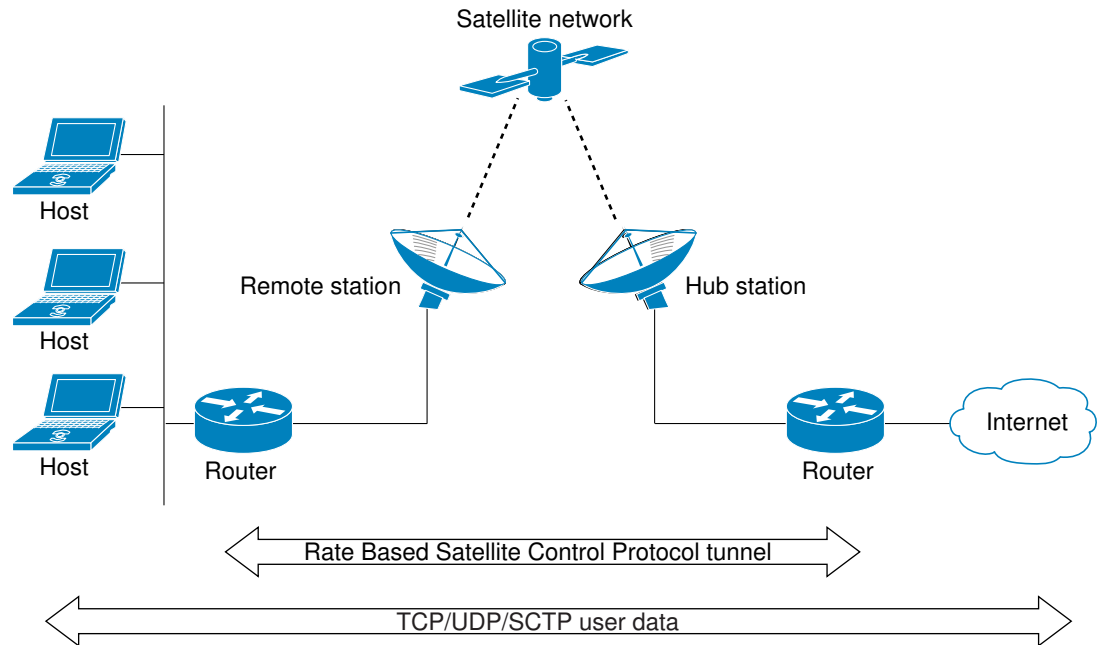
One of the disadvantages to using disruptive TCP PEP is the breaking of the end-to-end model. Some applications cannot work when the flow of traffic is broken, and the PEP has no provision for handling encrypted traffic (IPsec). New transport protocols such as SCTP require special handling or additional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

RBSCP 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 26 *Nondisruptive RBSCP Solution*



103374

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. This option should be used only when the RTT of the satellite link is greater than 700 milliseconds.
- **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.

Path MTU Discovery

Path MTU Discovery (PMTUD) can be enabled on a GRE or IP-in-IP tunnel interface. When PMTUD (RFC 1191) is enabled on a tunnel interface, the router performs PMTUD processing for the GRE (or IP-in-IP) tunnel IP packets. The router always performs PMTUD processing on the original data IP packets that enter the tunnel. When PMTUD is enabled, packet fragmentation is not permitted for packets that traverse the tunnel because the Don't Fragment (DF) bit is set on all the packets. If a packet that enters the tunnel encounters a link with a smaller MTU, the packet is dropped and an ICMP message is sent back to the sender of the packet. This message indicates that fragmentation was required (but not permitted) and provides the MTU of the link that caused the packet to be dropped.

For more detailed information about PMTUD, see the [IP Fragmentation and PMTUD](#) document.

**Note**

PMTUD on a tunnel interface requires that the tunnel endpoint be able to receive ICMP messages generated by routers in the path of the tunnel. Check that ICMP messages can be received before using PMTUD over firewall connections.

Use the **tunnel path-mtu-discovery** command to enable PMTUD for the tunnel packets, and use the **show interfaces tunnel** command to verify the tunnel PMTUD parameters. PMTUD currently works only on GRE and IP-in-IP tunnel interfaces.

QoS Options for Tunnels

A tunnel interface supports many of the same quality of service (QoS) features as a physical interface. QoS provides a way to ensure that mission-critical traffic has an acceptable level of performance. QoS options for tunnels include support for applying generic traffic shaping (GTS) directly on the tunnel interface and support for class-based shaping using the modular QoS CLI (MQC). Tunnel interfaces also support class-based policing, but they do not support committed access rate (CAR).

**Note**

Service policies are not supported on tunnel interfaces on the Cisco 7500 series routers.

GRE tunnels allow the router to copy the IP precedence bit values of the ToS byte to the tunnel or the GRE IP header that encapsulates the inner packet. Intermediate routers between the tunnel endpoints can use the IP precedence values to classify the packets for QoS features such as policy routing, weighted fair queuing (WFQ), and weighted random early detection (WRED).

When packets are encapsulated by tunnel or encryption headers, QoS features are unable to examine the original packet headers and correctly classify the packets. Packets that travel across the same tunnel have the same tunnel headers, so the packets are treated identically if the physical interface is congested. Tunnel packets can, however, be classified before tunneling and encryption can occur by using the QoS preclassify feature on the tunnel interface or on the crypto map.

**Note**

Class-based WFQ (CBWFQ) inside class-based shaping is not supported on a multipoint interface.

For examples of how to implement some QoS features on a tunnel interface, see the [Example Configuring QoS Options on Tunnel Interfaces](#), page 299.

How to Implement Tunnels

- [Determining the Tunnel Type, page 259](#)
- [Configuring a GRE Tunnel, page 261](#)
- [Configuring GRE IPv6 Tunnels, page 264](#)
- [Configuring GRE Tunnel IP Source and Destination VRF Membership, page 266](#)
- [Configuring a CTunnel, page 268](#)
- [Configuring GRE CLNS CTunnels to Carry IPv4 and IPv6 Packets, page 270](#)
- [Configuring Manual IPv6 Tunnels, page 272](#)
- [Configuring 6to4 Tunnels, page 274](#)
- [Configuring IPv4-Compatible IPv6 Tunnels, page 277](#)
- [Configuring ISATAP Tunnels, page 278](#)
- [Configuring the RBSCP Tunnel, page 280](#)
- [Verifying Tunnel Configuration and Operation, page 283](#)
- [Verifying RBSCP Tunnel Configuration and Operation, page 285](#)

Determining the Tunnel Type

Before configuring a tunnel, you must determine what type of tunnel you need to create.

SUMMARY STEPS

1. Determine the passenger protocol.
2. Determine the tunnel CLI type.
3. Determine the **tunnel mode** command keyword, if appropriate.

DETAILED STEPS

Step 1 Determine the passenger protocol.
The passenger protocol is the protocol that you are encapsulating.

Step 2 Determine the tunnel CLI type.
The table below shows how to determine the tunnel CLI command required for the transport protocol that you are using in the tunnel.

Table 19 *Determining the Tunnel CLI by the Transport Protocol*

Transport Protocol	Tunnel CLI Command
CLNS	ctunnel (with optional mode gre keywords)
Other	tunnel mode (with appropriate keyword)

Step 3 Determine the **tunnel mode** command keyword, if appropriate.
The table below shows how to determine the appropriate keyword to use with the **tunnel mode** command. In the tasks that follow in this module, only the relevant keywords for the **tunnel mode** command are displayed.

Table 20 *Determining the tunnel mode Command Keyword*

Keyword	Purpose
dvmrp	Use the dvmrp keyword to specify that the Distance Vector Multicast Routing Protocol encapsulation will be used.
gre ip	Use the gre ip keywords to specify that GRE encapsulation over IP will be used.
gre ipv6	Use the gre ipv6 keywords to specify that GRE encapsulation over IPv6 will be used.
gre multipoint	Use the gre multipoint keywords to specify that multipoint GRE (mGRE) encapsulation will be used.
ipip [decapsulate-any]	Use the ipip keyword to specify that IP-in-IP encapsulation will be used. The optional decapsulate-any keyword terminates any number of IP-in-IP tunnels at one tunnel interface. Note that this tunnel will not carry any outbound traffic; however, any number of remote tunnel endpoints can use a tunnel configured this way as their destination.
ipv6	Use the ipv6 keyword to specify that generic packet tunneling in IPv6 will be used.
ipv6ip	Use the ipv6ip keyword to specify that IPv6 will be used as the passenger protocol and IPv4 as both the carrier (encapsulation) and transport protocol. When additional keywords are not used, manual IPv6 tunnels are configured. Additional keywords can be used to specify IPv4-compatible, 6to4, or ISATAP tunnels.
mpls	Use the mpls keyword to specify that MPLS will be used for configuring Traffic Engineering (TE) tunnels.
rbscp	Use the rbscp keyword to specify that RBSCP tunnels will be used.

- [What to Do Next, page 260](#)

What to Do Next

To configure an RBSCP tunnel to carry IP data packets over a satellite or other long-distance delay link with high error rates, proceed to the [Configuring the RBSCP Tunnel, page 280](#).

Configuring a GRE Tunnel

Perform this task to configure a GRE tunnel. A tunnel interface is used to pass protocol traffic across a network that does not normally support the protocol. To build a tunnel, a tunnel interface must be defined on each of two routers and the tunnel interfaces must reference each other. At each router, the tunnel interface must be configured with a L3 address. The tunnel endpoints, tunnel source, and tunnel destination must be defined, and the type of tunnel must be selected. Optional steps can be performed to customize the tunnel.

Remember to configure the router at each end of the tunnel. If only one side of a tunnel is configured, the tunnel interface may still come up and stay up (unless keepalive is configured), but packets going into the tunnel will be dropped.

In Cisco IOS Release 12.2(8)T and later releases, CEF-switching over multipoint GRE tunnels was introduced. Previously, only process switching was available for multipoint GRE tunnels.

- [GRE Tunnel Keepalive, page 261](#)
- [What to Do Next, page 264](#)

GRE Tunnel Keepalive

Keepalive packets can be configured to be sent over IP-encapsulated GRE tunnels. You can specify the rate at which keepalives will be sent and the number of times that a device will continue to send keepalive packets without a response before the interface becomes inactive. GRE keepalive packets may be sent from both sides of a tunnel or from just one side.

Ensure that the physical interface to be used as the tunnel source in this task is up and configured with the appropriate IP address. For hardware technical descriptions and information about installing interfaces, see the hardware installation and configuration publication for your product.

**Note**

GRE tunnel keepalive is not supported in cases where virtual route forwarding (VRF) is applied to a GRE tunnel.

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **bandwidth** *kbps*
5. **keepalive** [*period* [*retries*]]
6. **tunnel source** {*ip-address* | *interface-type interface-number*}
7. **tunnel destination** {*hostname* | *ip-address*}
8. **tunnel key** *key-number*
9. **tunnel mode** {**gre ip** | **gre multipoint**}
10. **ip mtu** *bytes*
11. **ip tcp mss** *mss-value*
12. **tunnel path-mtu-discovery** [**age-timer** {*aging-mins* | **infinite**}]
13. **end**

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	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. <ul style="list-style-type: none"> • To configure a tunnel, use <i>tunnel</i> for the <i>type</i> argument. • On some router platforms such as the Cisco 7500 series, the number argument may consist of a slot, port adapter, and port number. For more details, see the interface command in the Cisco IOS Interface and Hardware Component Command Reference.
Step 4	bandwidth <i>kbps</i> Example: Router(config-if)# bandwidth 1000	Sets the current bandwidth value for an interface and communicates it to higher-level protocols. Specifies the tunnel bandwidth to be used to transmit packets. <ul style="list-style-type: none"> • Use the <i>kbps</i> argument to set the bandwidth, in kilobits per second (kbps). Note This is a routing parameter only; it does not affect the physical interface. The default bandwidth setting on a tunnel interface is 9.6 kbps. You should set the bandwidth on a tunnel to an appropriate value.

	Command or Action	Purpose
Step 5	<p>keepalive [<i>period</i> [<i>retries</i>]]</p> <p>Example:</p> <pre>Router(config-if)# keepalive 3 7</pre>	<p>(Optional) Specifies the number of times that the device will continue to send keepalive packets without response before bringing the tunnel interface protocol down.</p> <ul style="list-style-type: none"> GRE keepalive packets may be configured either on only one side of the tunnel or on both. If GRE keepalive is configured on both sides of the tunnel, the <i>period</i> and <i>retries</i> arguments can be different at each side of the link. <p>Note This command is supported only on GRE point-to-point tunnels.</p>
Step 6	<p>tunnel source {<i>ip-address</i> <i>interface-type interface-number</i>}</p> <p>Example:</p> <pre>Router(config-if)# tunnel source Ethernet 1</pre>	<p>Configures the tunnel source.</p> <ul style="list-style-type: none"> Use the <i>ip-address</i> argument to specify the source IP address. Use the <i>interface-type</i> and <i>interface-number</i> arguments to specify the interface to use. <p>Note The tunnel source and destination IP addresses must be defined on two separate devices.</p>
Step 7	<p>tunnel destination {<i>hostname</i> <i>ip-address</i>}</p> <p>Example:</p> <pre>Router(config-if)# tunnel destination 172.17.2.1</pre>	<p>Configures the tunnel destination.</p> <ul style="list-style-type: none"> Use the <i>hostname</i> argument to specify the name of the host destination. Use the <i>ip-address</i> argument to specify the IP address of the host destination. <p>Note The tunnel source and destination IP addresses must be defined on two separate devices.</p>
Step 8	<p>tunnel key <i>key-number</i></p> <p>Example:</p> <pre>Router(config-if)# tunnel key 1000</pre>	<p>(Optional) Enables an ID key for a tunnel interface.</p> <ul style="list-style-type: none"> Use the <i>key-number</i> argument to identify a tunnel key that is carried in each packet. Tunnel ID keys can be used as a form of weak security to prevent improper configuration or injection of packets from a foreign source. <p>Note This command is supported only on GRE tunnel interfaces. We do not recommend relying on this key for security purposes.</p>
Step 9	<p>tunnel mode {gre ip gre multipoint}</p> <p>Example:</p> <pre>Router(config-if)# tunnel mode gre ip</pre>	<p>Specifies the encapsulation protocol to be used in the tunnel.</p> <ul style="list-style-type: none"> Use the gre ip keywords to specify that GRE over IP encapsulation will be used. Use the gre multipoint keywords to specify that multipoint GRE (mGRE) will be used.

Command or Action	Purpose
<p>Step 10 <code>ip mtu bytes</code></p> <p>Example:</p> <pre>Router(config-if)# ip mtu 1400</pre>	<p>(Optional) Set the maximum transmission unit (MTU) size of IP packets sent on an interface.</p> <ul style="list-style-type: none"> If an IP packet exceeds the MTU set for the interface, the Cisco IOS software will fragment it unless the DF bit is set. All devices on a physical medium must have the same protocol MTU in order to operate. <p>Note If the tunnel path-mtu-discovery command is enabled in Step 12, do not configure this command.</p>
<p>Step 11 <code>ip tcp mss mss-value</code></p> <p>Example:</p> <pre>Router(config-if)# ip tcp mss 250</pre>	<p>(Optional) Specifies the maximum segment size (MSS) for TCP connections that originate or terminate on a router.</p> <ul style="list-style-type: none"> Use the <i>mss-value</i> argument to specify the maximum segment size for TCP connections, in bytes.
<p>Step 12 <code>tunnel path-mtu-discovery [aging-timer {aging-mins infinite}]</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel path-mtu-discovery</pre>	<p>(Optional) Enables Path MTU Discovery (PMTUD) on a GRE or IP-in-IP tunnel interface.</p> <ul style="list-style-type: none"> When PMTUD is enabled on a tunnel interface, PMTUD will operate for GRE IP tunnel packets to minimize fragmentation in the path between the tunnel endpoints.
<p>Step 13 <code>end</code></p> <p>Example:</p> <pre>Router(config-if)# end</pre>	<p>Exits interface configuration mode and returns to privileged EXEC mode.</p>

What to Do Next

Proceed to the [Verifying Tunnel Configuration and Operation](#), page 283.

Configuring GRE IPv6 Tunnels

Perform this task to configure a GRE tunnel on an IPv6 network. GRE tunnels can be configured to run over an IPv6 network layer and transport IPv6 and IPv4 packets through IPv6 tunnels.

When GRE IPv6 tunnels are configured, IPv6 addresses are assigned to the tunnel source and the tunnel destination. The tunnel interface can have either IPv4 or IPv6 addresses (this is not shown in the task below). The host or device at each end of the configured tunnel must support both IPv4 and IPv6 protocol stacks.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *tunnel-number*
4. **tunnel source** {*ipv6-address* | *interface-type interface-number*}
5. **tunnel destination** *ipv6-address*
6. **tunnel mode gre ipv6**
7. **end**

DETAILED STEPS

Command or Action	Purpose
Step 1 enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2 configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3 interface tunnel <i>tunnel-number</i> Example: Device(config)# interface tunnel 0	Specifies a tunnel interface and number and enters interface configuration mode.
Step 4 tunnel source { <i>ipv6-address</i> <i>interface-type interface-number</i> } Example: Device(config-if)# tunnel source ethernet 0	Specifies the source IPv6 address or the source interface type and number for the tunnel interface. <ul style="list-style-type: none"> • If an interface type and number are specified, the interface must be configured with an IPv6 address. Note Only the syntax used in this context is displayed. For more details, see the IPv6 Command Reference .
Step 5 tunnel destination <i>ipv6-address</i> Example: Device(config-if)# tunnel destination 2001:0DB8:0C18:2::300	Specifies the destination IPv6 address for the tunnel interface. Note Only the syntax used in this context is displayed. For more details, see the IPv6 Command Reference .

Command or Action	Purpose
Step 6 <code>tunnel mode gre ipv6</code> Example: Device(config-if)# tunnel mode gre ipv6	Specifies a GRE IPv6 tunnel. Note The tunnel mode gre ipv6 command specifies GRE as the encapsulation protocol for the tunnel interface. Only the syntax used in this context is displayed. For more details, see the IPv6 Command Reference .
Step 7 <code>end</code> Example: Device(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.

- [What to Do Next, page 266](#)

What to Do Next

Proceed to the “Verifying Tunnel Configuration and Operation” section.

Configuring GRE Tunnel IP Source and Destination VRF Membership

This task explains how to configure the source and destination of a tunnel to belong to any virtual private network (VPN) routing/forwarding (VRFs) tables



Note

Cisco 10000 Series Routers

- The VRF associated with the **tunnel vrf** command is the same as the VRF associated with the physical interface over which the tunnel sends packets (outer IP packet routing).
- The VRF associated with the tunnel by using the **ip vrf forwarding** command is the VRF that the packets are to be forwarded in as the packets exit the tunnel (inner IP packet routing).
- The Cisco 10000 series router does not support the fragmentation of multicast packets passing through a multicast tunnel.

>

SUMMARY STEPS

1. **enable**
2. **configure** { **terminal** | **memory** | **network** }
3. **interface tunnel** *slot*
4. **ip vrf forwarding** *vrf-name*
5. **ip address** *ip-address subnet-mask*
6. **tunnel source** { *ip-address* | *type number* }
7. **tunnel destination** { *hostname* | *ip-address* }
8. **tunnel vrf** *vrf-name*

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 <code>enable</code></p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables higher privilege levels, such as privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
<p>Step 2 <code>configure {terminal memory network}</code></p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
<p>Step 3 <code>interface tunnel slot</code></p> <p>Example:</p> <pre>Router(config)# interface tunnel 0</pre>	<p>Enters interface configuration mode for the specified interface.</p>
<p>Step 4 <code>ip vrf forwarding vrf-name</code></p> <p>Example:</p> <pre>Router(config-if)# ip vrf forwarding green</pre>	<p>Defines the VRF.</p>
<p>Step 5 <code>ip address ip-address subnet-mask</code></p> <p>Example:</p> <pre>Router(config-if)# ip address 10.7.7.7 255.255.255.255</pre>	<p>Specifies the ip address and subnet mask.</p>
<p>Step 6 <code>tunnel source {ip-address type number}</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel source loop 0</pre>	<p>Specifies the tunnel source.</p>
<p>Step 7 <code>tunnel destination {hostname ip-address}</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel destination 10.5.5.5</pre>	<p>Defines the tunnel destination.</p>

Command or Action	Purpose
Step 8 <code>tunnel vrf <i>vrf-name</i></code> Example: <code>Router(config-if)# tunnel vrf financel</code>	Defines the VRF.

- [What to Do Next, page 268](#)

What to Do Next

Proceed to the [Verifying Tunnel Configuration and Operation, page 283](#).

Configuring a CTunnel

Perform this task to configure an IP over CLNS tunnel (CTunnel). To configure a CTunnel between a single pair of routers, a tunnel interface must be configured with an IP address, and a tunnel destination must be defined. The destination network service access point (NSAP) address for Router A would be the NSAP address of Router B, and the destination NSAP address for Router B would be the NSAP address of Router A. Ideally, the IP addresses used for the virtual interfaces at either end of the tunnel should be in the same IP subnet. Remember to configure the router at each end of the tunnel.

- [CTunnel, page 268](#)
- [Troubleshooting Tips, page 270](#)
- [What to Do Next, page 270](#)

CTunnel

A CTunnel lets you transport IP traffic over Connectionless Network Service (CLNS), for example, on the data communications channel (DCC) of a SONET ring. CTunnels allow IP packets to be tunneled through the Connectionless Network Protocol (CLNP) to preserve TCP/IP services.

Configuring a CTunnel allows you to telnet to a remote router that has only CLNS connectivity. Other management facilities can also be used, such as Simple Network Management Protocol (SNMP) and TFTP, which otherwise would not be available over a CLNS network.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface ctunnel** *interface-number*
4. **ip address** *ip-address mask*
5. **ctunnel destination** *remote-nsap-address*
6. **end**
7. **show interfaces ctunnel** *interface-number*

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 <code>enable</code></p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
<p>Step 2 <code>configure terminal</code></p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
<p>Step 3 <code>interface ctunnel interface-number</code></p> <p>Example:</p> <pre>Router(config)# interface ctunnel 102</pre>	<p>Creates a virtual interface to transport IP over a CLNS tunnel and enters interface configuration mode.</p> <p>Note The interface number must be unique for each CTunnel interface.</p>
<p>Step 4 <code>ip address ip-address mask</code></p> <p>Example:</p> <pre>Router(config-if)# ip address 10.0.0.1 255.255.255.0</pre>	<p>Enables IP on the interface.</p> <ul style="list-style-type: none"> Use the <i>ip-address</i> and <i>mask</i> arguments to specify the IP address and mask for the interface.
<p>Step 5 <code>ctunnel destination remote-nsap-address</code></p> <p>Example:</p> <pre>Router(config-if)# ctunnel destination 49.0001.2222.2222.2222.00</pre>	<p>Specifies the destination NSAP address of the CTunnel, where the packets exit the tunnel.</p> <ul style="list-style-type: none"> Use the <i>remote-nsap-address</i> argument to specify the NSAP address at the CTunnel endpoint.
<p>Step 6 <code>end</code></p> <p>Example:</p> <pre>Router(config-if)# end</pre>	<p>Exits interface configuration mode and returns to privileged EXEC mode.</p>
<p>Step 7 <code>show interfaces ctunnel interface-number</code></p> <p>Example:</p> <pre>Router# show interfaces ctunnel 102</pre>	<p>(Optional) Displays information about an IP over CLNS tunnel.</p> <ul style="list-style-type: none"> Use the <i>interface-number</i> argument to specify a CTunnel interface. Use this command to verify the CTunnel configuration.

Troubleshooting Tips

Use the **ping** command to diagnose basic network connectivity issues.

What to Do Next

Proceed to the [Verifying Tunnel Configuration and Operation, page 283](#).

Configuring GRE CLNS CTunnels to Carry IPv4 and IPv6 Packets

Perform this task to configure a CTunnel in GRE mode to transport IPv4 and IPv6 packets in a CLNS network.

To configure a CTunnel between a single pair of routers, a tunnel interface must be configured with an IP address, and a tunnel destination must be defined. The destination network service access point (NSAP) address for Router A would be the NSAP address of Router B, and the destination NSAP address for Router B would be the NSAP address of Router A. Ideally, the IP addresses used for the virtual interfaces at either end of the tunnel should be in the same IP subnet. Remember to configure the router at each end of the tunnel.

- [Tunnels for IPv4 and IPv6 Packets over CLNS Networks, page 270](#)
- [What to Do Next, page 272](#)

Tunnels for IPv4 and IPv6 Packets over CLNS Networks

Configuring the **ctunnel mode gre** command on a CTunnel interface enables IPv4 and IPv6 packets to be tunneled over CLNS in accordance with RFC 3147. Compliance with this RFC should allow interoperation between Cisco equipment and that of other vendors in which the same standard is implemented.

RFC 3147 specifies the use of GRE for tunneling packets. The implementation of this feature does not include support for GRE services defined in header fields, such as those used to specify checksums, keys, or sequencing. Any packets received that specify the use of these features will be dropped.

The default CTunnel mode continues to use the standard Cisco encapsulation, which will tunnel only IPv4 packets. If you want to tunnel IPv6 packets, you must use the GRE encapsulation mode. Both ends of the tunnel must be configured with the same mode for either method to work.

- An IPv4 or IPv6 address must be configured on a CTunnel interface, and manually configured CLNS addresses must be assigned to the CTunnel destination.
- The host or router at each end of a configured CTunnel must support both the IPv4 and IPv6 protocol stacks.
- The CTunnel source and destination must both be configured to run in the same mode.



Note

GRE services, such as those used to specify checksums, keys, or sequencing, are not supported. Packets that request use of those features will be dropped.

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface ctunnel** *interface-number*
4. Do one of the following:
 - **ip address** *ip-address mask*
 -
 - **ipv6 address** *ipv6-prefix / prefix-length [eui-64]*
5. **ctunnel destination** *remote-nsap-address*
6. **ctunnel mode gre**
7. **end**
8. **show interfaces ctunnel** *interface-number*

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 enable</p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
<p>Step 2 configure terminal</p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
<p>Step 3 interface ctunnel <i>interface-number</i></p> <p>Example:</p> <pre>Router(config)# interface ctunnel 102</pre>	<p>Creates a virtual interface to transport IP over a CLNS tunnel and enters interface configuration mode.</p> <p>Note The interface number must be unique for each CTunnel interface.</p>
<p>Step 4 Do one of the following:</p> <ul style="list-style-type: none"> • ip address <i>ip-address mask</i> • • ipv6 address <i>ipv6-prefix / prefix-length [eui-64]</i> <p>Example:</p> <pre>Router(config-if)# ipv6 address 2001:0DB8:1234:5678::3/126</pre>	<p>Specifies the IPv4 or IPv6 network assigned to the interface and enables IPv4 or IPv6 packet processing on the interface.</p> <p>Note For more information about IPv6 network, see the "Configuring Basic Connectivity for IPv6" module in the Cisco IOS IPv6 Configuration Guide.</p>

Command or Action	Purpose
<p>Step 5 <code>ctunnel destination</code> <i>remote-nsap-address</i></p> <p>Example:</p> <pre>Router(config-if)# ctunnel destination 192.168.30.1</pre>	<p>Specifies the destination NSAP address of the CTunnel, where the packets are extracted.</p> <ul style="list-style-type: none"> Use the <i>remote-nsap-address</i> argument to specify the NSAP address at the CTunnel endpoint.
<p>Step 6 <code>ctunnel mode gre</code></p> <p>Example:</p> <pre>Router(config-if)# ctunnel mode gre</pre>	<p>Specifies a CTunnel running in GRE mode for both IPv4 and IPv6 traffic.</p> <p>Note The <code>ctunnel mode gre</code> command specifies GRE as the encapsulation protocol for the tunnel.</p>
<p>Step 7 <code>end</code></p> <p>Example:</p> <pre>Router(config-if)# end</pre>	<p>Exits interface configuration mode and returns to privileged EXEC mode.</p>
<p>Step 8 <code>show interfaces ctunnel</code> <i>interface-number</i></p> <p>Example:</p> <pre>Router# show interfaces ctunnel 102</pre>	<p>(Optional) Displays information about an IP over CLNS tunnel.</p> <ul style="list-style-type: none"> Use the <i>interface-number</i> argument to specify a CTunnel interface. Use this command to verify the CTunnel configuration.

What to Do Next

Proceed to the [Verifying Tunnel Configuration and Operation, page 283](#).

Configuring Manual IPv6 Tunnels

This task explains how to configure a manual IPv6 overlay tunnel.

With manually configured IPv6 tunnels, an IPv6 address is configured on a tunnel interface and manually configured IPv4 addresses are assigned to the tunnel source and the tunnel destination. The host or router at each end of a configured tunnel must support both the IPv4 and IPv6 protocol stacks.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *tunnel-number*
4. **ipv6 address** *ipv6-prefix / prefix-length* [**eui-64**]
5. **tunnel source** {*ip-address*| *interface-type interface-number*}
6. **tunnel destination** *ip-address*
7. **tunnel mode ipv6ip**
8. **end**

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 enable</p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
<p>Step 2 configure terminal</p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
<p>Step 3 interface tunnel <i>tunnel-number</i></p> <p>Example:</p> <pre>Router(config)# interface tunnel 0</pre>	<p>Specifies a tunnel interface and number and enters interface configuration mode.</p>
<p>Step 4 ipv6 address <i>ipv6-prefix / prefix-length</i> [eui-64]</p> <p>Example:</p> <pre>Router(config-if)# ipv6 address 2001:0DB8:1234:5678::3/126</pre>	<p>Specifies the IPv6 network assigned to the interface and enables IPv6 processing on the interface.</p> <p>Note For more information on configuring IPv6 addresses, see the "Configuring Basic Connectivity for IPv6" module.</p>
<p>Step 5 tunnel source {<i>ip-address</i> <i>interface-type interface-number</i>}</p> <p>Example:</p> <pre>Router(config-if)# tunnel source ethernet 0</pre>	<p>Specifies the source IPv4 address or the source interface type and number for the tunnel interface.</p> <ul style="list-style-type: none"> • If an interface is specified, the interface must be configured with an IPv4 address.

Command or Action	Purpose
<p>Step 6 <code>tunnel destination ip-address</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel destination 192.168.30.1</pre>	Specifies the destination IPv4 address for the tunnel interface.
<p>Step 7 <code>tunnel mode ipv6ip</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel mode ipv6ip</pre>	<p>Specifies a manual IPv6 tunnel.</p> <p>Note The <code>tunnel mode ipv6ip</code> command specifies IPv6 as the passenger protocol and IPv4 as both the carrier (encapsulation) and transport protocol for the manual IPv6 tunnel.</p>
<p>Step 8 <code>end</code></p> <p>Example:</p> <pre>Router(config-if)# end</pre>	Exits interface configuration mode and returns to privileged EXEC mode.

- [What to Do Next, page 274](#)

What to Do Next

Proceed to the [Verifying Tunnel Configuration and Operation, page 283](#).

Configuring 6to4 Tunnels

This task explains how to configure a 6to4 overlay tunnel.

With 6to4 tunnels, the tunnel destination is determined by the border-router IPv4 address, which is concatenated to the prefix 2002::/16 in the format 2002:border-router-IPv4-address::/48. The border router at each end of a 6to4 tunnel must support both the IPv4 and IPv6 protocol stacks.



Note

The configuration of only one IPv4-compatible tunnel and one 6to4 IPv6 tunnel is supported on a router. If you choose to configure both of these tunnel types on the same router, we strongly recommend that they not share the same tunnel source.

The reason that a 6to4 tunnel and an IPv4-compatible tunnel cannot share the same interface is that both of them are NBMA "point-to-multipoint" access links and only the tunnel source can be used to reorder the packets from a multiplexed packet stream into a single packet stream for an incoming interface. So when a packet with an IPv4 protocol type of 41 arrives on an interface, that packet is mapped to an IPv6 tunnel interface on the basis of the IPv4 address. However, if both the 6to4 tunnel and the IPv4-compatible tunnel share the same source interface, the router cannot determine the IPv6 tunnel interface to which it should assign the incoming packet.

IPv6 manually configured tunnels can share the same source interface because a manual tunnel is a "point-to-point" link, and both the IPv4 source and IPv4 destination of the tunnel are defined.

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel *tunnel-number***
4. **ipv6 address *ipv6-prefix / prefix-length [eui-64]***
5. **tunnel source {*ip-address*| *interface-type interface-number*}**
6. **tunnel mode ipv6ip 6to4**
7. **exit**
8. **ipv6 route *ipv6-prefix / prefix-length tunnel tunnel-number***

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 enable</p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
<p>Step 2 configure terminal</p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>

Command or Action	Purpose
<p>Step 3 <code>interface tunnel tunnel-number</code></p> <p>Example:</p> <pre>Router(config)# interface tunnel 0</pre>	Specifies a tunnel interface and number and enters interface configuration mode.
<p>Step 4 <code>ipv6 address ipv6-prefix / prefix-length [eui-64]</code></p> <p>Example:</p> <pre>Router(config-if)# ipv6 address 2002:c0a8:6301:1::1/64</pre>	<p>Specifies the IPv6 address assigned to the interface and enables IPv6 processing on the interface.</p> <ul style="list-style-type: none"> The 32 bits following the initial 2002::/16 prefix correspond to an IPv4 address assigned to the tunnel source. <p>Note For more information about configuring IPv6 addresses, see the "Configuring Basic Connectivity for IPv6" module.</p>
<p>Step 5 <code>tunnel source {ip-address interface-type interface-number}</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel source ethernet 0</pre>	<p>Specifies the source IPv4 address or the source interface type and number for the tunnel interface.</p> <p>Note The interface type and number specified in the tunnel source command must be configured with an IPv4 address.</p>
<p>Step 6 <code>tunnel mode ipv6ip 6to4</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel mode ipv6ip 6to4</pre>	Specifies an IPv6 overlay tunnel using a 6to4 address.
<p>Step 7 <code>exit</code></p> <p>Example:</p> <pre>Router(config-if)# exit</pre>	Exits interface configuration mode and returns to global configuration mode.
<p>Step 8 <code>ipv6 route ipv6-prefix / prefix-length tunnel tunnel-number</code></p> <p>Example:</p> <pre>Router(config)# ipv6 route 2002::/16 tunnel 0</pre>	<p>Configures a static route for the IPv6 6to4 prefix 2002::/16 to the specified tunnel interface.</p> <p>Note When configuring a 6to4 overlay tunnel, you must configure a static route for the IPv6 6to4 prefix 2002::/16 to the 6to4 tunnel interface.</p> <ul style="list-style-type: none"> The tunnel number specified in the ipv6 route command must be the same tunnel number specified in the interface tunnel command.

- [What to Do Next, page 277](#)

What to Do Next

Proceed to the [Verifying Tunnel Configuration and Operation](#), page 283.

Configuring IPv4-Compatible IPv6 Tunnels

This task explains how to configure an IPv4-compatible IPv6 overlay tunnel.

With an IPv4-compatible tunnel, the tunnel destination is automatically determined by the IPv4 address in the low-order 32 bits of IPv4-compatible IPv6 addresses. The host or router at each end of an IPv4-compatible tunnel must support both the IPv4 and IPv6 protocol stacks.



Note

IPv4-compatible tunnels were initially supported for IPv6, but Cisco now recommends that you use a different IPv6 overlay tunneling technique.

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *tunnel-number*
4. **tunnel source** {*ip-address*| *interface-type interface-number*}
5. **tunnel mode ipv6ip auto-tunnel**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	interface tunnel <i>tunnel-number</i> Example: Router(config)# interface tunnel 0	Specifies a tunnel interface and number and enters interface configuration mode.

Command or Action	Purpose
<p>Step 4 tunnel source {<i>ip-address</i> <i>interface-type interface-number</i>}</p> <p>Example:</p> <pre>Router(config-if)# tunnel source ethernet 0</pre>	<p>Specifies the source IPv4 address or the source interface type and number for the tunnel interface.</p> <p>Note The interface type and number specified in the tunnel source command must be configured with an IPv4 address.</p>
<p>Step 5 tunnel mode ipv6ip auto-tunnel</p> <p>Example:</p> <pre>Router(config-if)# tunnel mode ipv6ip auto-tunnel</pre>	<p>Specifies an IPv4-compatible tunnel using an IPv4-compatible IPv6 address.</p>

- [What to Do Next, page 278](#)

What to Do Next

Proceed to the [Verifying Tunnel Configuration and Operation, page 283](#).

Configuring ISATAP Tunnels

This task describes how to configure an ISATAP overlay tunnel.

The **tunnel source** command used in the configuration of an ISATAP tunnel must point to an interface that is configured with an IPv4 address. The ISATAP IPv6 address and prefix (or prefixes) advertised are configured for a native IPv6 interface. The IPv6 tunnel interface must be configured with a modified EUI-64 address because the last 32 bits in the interface identifier are constructed using the IPv4 tunnel source address.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *tunnel-number*
4. **ipv6 address** *ipv6-prefix prefix-length* [**eui-64**]
5. **no ipv6 nd suppress-ra**
6. **tunnel source** {*ip-address* | *interface-type interface-number*}
7. **tunnel mode ipv6ip isatap**
8. **end**

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 <code>enable</code></p> <p>Example: Device> enable</p>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
<p>Step 2 <code>configure terminal</code></p> <p>Example: Device# configure terminal</p>	<p>Enters global configuration mode.</p>
<p>Step 3 <code>interface tunnel tunnel-number</code></p> <p>Example: Device(config)# interface tunnel 1</p>	<p>Specifies a tunnel interface and number and enters interface configuration mode.</p>
<p>Step 4 <code>ipv6 address ipv6-prefix prefix-length [eui-64]</code></p> <p>Example: Device(config-if)# ipv6 address 2001:0DB8:6301::/64 eui-64</p>	<p>Specifies the IPv6 address assigned to the interface and enables IPv6 processing on the interface.</p> <p>Note For more information on configuring IPv6 addresses, see the "Configuring Basic Connectivity for IPv6" module.</p>
<p>Step 5 <code>no ipv6 nd suppress-ra</code></p> <p>Example: Device(config-if)# no ipv6 nd suppress-ra</p>	<p>Enables sending of IPv6 device advertisements to allow client autoconfiguration.</p> <ul style="list-style-type: none"> Sending of IPv6 device advertisements is disabled by default on tunnel interfaces.
<p>Step 6 <code>tunnel source {ip-address interface-type interface-number}</code></p> <p>Example: Device(config-if)# tunnel source GigabitEthernet 1/0/1</p>	<p>Specifies the source IPv4 address or the source interface type and number for the tunnel interface.</p> <p>Note The interface type and number specified in the tunnel source command must be configured with an IPv4 address.</p>
<p>Step 7 <code>tunnel mode ipv6ip isatap</code></p> <p>Example: Device(config-if)# tunnel mode ipv6ip isatap</p>	<p>Specifies an IPv6 overlay tunnel using an ISATAP address.</p>
<p>Step 8 <code>end</code></p> <p>Example: Device(config-if)# end</p>	<p>Exits interface configuration mode and returns to privileged EXEC mode.</p>

- [What to Do Next](#), page 280

What to Do Next

Proceed to the [Verifying Tunnel Configuration and Operation](#), page 283.

Configuring the RBSCP Tunnel

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

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



Note

- 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 control 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.

>

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 rbscp**
9. **tunnel rbscp ack-split** *split-size*
10. **tunnel rbscp delay**
11. **tunnel rbscp report**
12. **tunnel rbscp window-stuff** *step-size*
13. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>interface <i>type number</i></p> <p>Example:</p> <pre>Router(config)# interface tunnel 0</pre>	<p>Specifies the interface type and number and enters interface configuration mode.</p>
Step 4	<p>ip unnumbered <i>interface-type interface-number</i></p> <p>Example:</p> <pre>Router(config-if)# ip unnumbered Ethernet 1</pre>	<p>Enables IP processing on an interface without assigning an explicit IP address.</p> <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	<p>tunnel source {<i>ip-address</i> <i>interface-type interface-number</i>}</p> <p>Example:</p> <pre>Router(config-if)# tunnel source Ethernet 1</pre>	<p>Configures the tunnel source.</p> <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 Cisco recommends specifying an interface as the tunnel source.
Step 6	<p>tunnel destination {<i>hostname</i> <i>ip-address</i>}</p> <p>Example:</p> <pre>Router(config-if)# tunnel destination 172.17.2.1</pre>	<p>Configures the tunnel destination.</p> <ul style="list-style-type: none"> Use the <i>hostname</i> argument to specify the name of the host destination. Use the <i>ip-address</i> argument to specify the IP address of the host destination.

Command or Action	Purpose
<p>Step 7 <code>tunnel bandwidth {receive transmit} bandwidth</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel bandwidth transmit 1000</pre>	<p>Specifies the tunnel bandwidth to be used to transmit packets.</p> <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>
<p>Step 8 <code>tunnel mode rbscp</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel mode rbscp</pre>	<p>Specifies the protocol to be used in the tunnel.</p> <ul style="list-style-type: none"> Use the rbscp keyword to specify that RBSCP will be used as the tunnel protocol.
<p>Step 9 <code>tunnel rbscp ack-split split-size</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel rbscp ack-split 6</pre>	<p>(Optional) Enables TCP acknowledgement (ACK) splitting with RBSCP tunnels.</p> <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.
<p>Step 10 <code>tunnel rbscp delay</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel rbscp delay</pre>	<p>(Optional) Enables RBSCP tunnel delay.</p> <ul style="list-style-type: none"> Use this command only when the RTT measured between the two routers nearest to the satellite links is greater than 700 milliseconds.
<p>Step 11 <code>tunnel rbscp report</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel rbscp report</pre>	<p>(Optional) Reports dropped RBSCP packets to SCTP.</p> <ul style="list-style-type: none"> Reporting dropped packets to SCTP provides better bandwidth use because RBSCP tells the SCTP implementation at the end hosts to retransmit the dropped packets and this prevents the end hosts from assuming that the network is congested.
<p>Step 12 <code>tunnel rbscp window-stuff step-size</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel rbscp window-stuff 1</pre>	<p>(Optional) Enables TCP window stuffing by increasing the value of the TCP window scale for RBSCP tunnels.</p> <ul style="list-style-type: none"> Use the <i>step-size</i> argument to specify the step increment number.
<p>Step 13 <code>end</code></p> <p>Example:</p> <pre>Router(config-if)# end</pre>	<p>Exits interface configuration mode and returns to privileged EXEC mode.</p>

- [What to Do Next, page 283](#)

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, hostnames, and other parameters for the appropriate values on the second router.

After the task is completed on the router on the other side of the satellite link, proceed to the [Verifying RBSCP Tunnel Configuration and Operation, page 285](#).

Verifying Tunnel Configuration and Operation

This optional task explains how to verify tunnel configuration and operation. The commands contained in the task steps can be used in any sequence and may need to be repeated. The following commands can be used for GRE tunnels, IPv6 manually configured tunnels, and IPv6 over IPv4 GRE tunnels. This process includes the following general steps (details follow):

- On Router A, ping the IP address of the CTunnel interface of Router B.
- On Router B, ping the IP address of the CTunnel interface of Router A.

SUMMARY STEPS

1. **enable**
2. **show interfaces tunnel** *number* [**accounting**]
3. **ping** [*protocol*] *destination*
4. **show ip route** [*address*]*[mask]*
5. **ping** [*protocol*] *destination*

DETAILED STEPS

Step 1

enable

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Router> enable
```

Step 2

show interfaces tunnel *number* [**accounting**]

Assuming a generic example suitable for both IPv6 manually configured tunnels and IPv6 over IPv4 GRE tunnels, two routers are configured to be endpoints of a tunnel. Router A has Ethernet interface 0/0 configured as the source for tunnel interface 0 with an IPv4 address of 10.0.0.1 and an IPv6 prefix of 2001:0DB8:1111:2222::1/64. Router B has Ethernet interface 0/0 configured as the source for tunnel interface 1 with an IPv4 address of 10.0.0.2 and an IPv6 prefix of 2001:0DB8:1111:2222::2/64.

To verify that the tunnel source and destination addresses are configured, use the **show interfaces tunnel** command on Router A.

Example:

```
RouterA# show interfaces tunnel 0
```

```
Tunnel0 is up, line protocol is up
Hardware is Tunnel
MTU 1514 bytes, BW 9 Kbit, DLY 500000 usec,
  reliability 255/255, txload 1/255, rxload 1/255
Encapsulation TUNNEL, loopback not set
Keepalive not set
Tunnel source 10.0.0.1 (Ethernet0/0), destination 10.0.0.2, fastswitch TTL 255
Tunnel protocol/transport GRE/IP, key disabled, sequencing disabled
Tunnel TTL 255
Checksumming of packets disabled, fast tunneling enabled
Last input 00:00:14, output 00:00:04, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue :0/0 (size/max)
 5 minute input rate 0 bits/sec, 0 packets/sec
 5 minute output rate 0 bits/sec, 0 packets/sec
 4 packets input, 352 bytes, 0 no buffer
 Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
 0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
 8 packets output, 704 bytes, 0 underruns
 0 output errors, 0 collisions, 0 interface resets
 0 output buffer failures, 0 output buffers swapped out
```

Step 3 ping [protocol] destination

To check that the local endpoint is configured and working, use the **ping** command on Router A.

Example:

```
RouterA# ping
 2001:0DB8:1111:2222::2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:0DB8:1111:2222::2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/20/20 ms
```

Step 4 show ip route [address[mask]]

To check that a route exists to the remote endpoint address, use the **show ip route** command.

Example:

```
RouterA# show ip route 10.0.0.2
Routing entry for 10.0.0.0/24
  Known via "connected", distance 0, metric 0 (connected, via interface)
  Routing Descriptor Blocks:
    * directly connected, via Ethernet0/0
      Route metric is 0, traffic share count is 1
```

Step 5 ping [protocol] destination

To check that the remote endpoint address is reachable, use the **ping** command on Router A.

Note The remote endpoint address may not be reachable using the **ping** command because of filtering, but the tunnel traffic may still reach its destination.

Example:

```
RouterA# ping 10.0.0.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.0.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/21/28 ms
```

To check that the remote IPv6 tunnel endpoint is reachable, use the **ping** command again on Router A. The same note on filtering also applies to this example.

Example:

```
RouterA# ping 1::2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1::2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/20/20 ms
```

These steps may be repeated at the other endpoint of the tunnel.

Verifying RBSCP Tunnel Configuration and Operation

Perform one or both of the following optional tasks to verify the configuration and operation of the RBSCP tunnel configured in the [Configuring the RBSCP Tunnel, page 280](#).

- [Verifying That the RBSCP Tunnel Is Active, page 285](#)
- [Verifying the RBSCP Traffic, page 286](#)

Verifying That the RBSCP Tunnel Is Active

Perform this task to verify that the RBSCP tunnel is active.

SUMMARY STEPS

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

DETAILED STEPS

Step 1**enable**

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Router> enable
```

Step 2**show rbscp [all| state| statistics] [tunnel tunnel-number]**

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.

Example:

```
Router# show rbscp state tunnel 1
Tunnell is up, line protocol is up
RBSCP operational state: OPEN
```

```

RBSCP operating mode: (264h) ack-split window-stuffing inorder SCTP-report
window step: 1
drop scale: 0
ACK split size: 4
input drop scale: 2
initial TSN: 1h
fuzz factor: 0
max burst: tunnel 0, network 0
next TSN: 1h
next sequence: 16Bh
current outstanding: 0
max out per RTT: 2062500
packets since SACK: 0
cumulative ack: 0h
TSN at SACK: 0h
last cumulative ack: 0h
last delivered TSN: 0h
next FWDTSN corr: 0h
RTO: 704 ms
RTT: 550 ms      srtt_sa: 4391      srtt_sv: 3
sentQ: num packets: 0, num bytes: 0
tmitQ: num packets: 0, num bytes: 0

```

Use this command with the **statistics** and **tunnel** keywords to display statistical information about the tunnel. All counters display totals accumulated since the last **clear rbscp** command was issued.

Example:

```

Router# show rbscp statistics tunnel 0
Tunnel0 is up, line protocol is up
RBSCP protocol statistics:
Init FWD-TSNs sent 0, received 0
TUNNEL-UPs sent 0, received 0
CLOSEDs sent 0, received 0
TSNs sent 0, resent 0, lost by sender 0
TSNs received 0 (duplicates 0)
FWD-TSNs sent 144 (heartbeats 0)
FWD-TSNs received 0 (ignored 0)
FWD-TSNs caused 0 packet drops, 0 whole window drops
SACKs sent 0, received 0 (ignored 0)
Recovered with RTX 0
Received with delay 0
Most released at once 0
Failed sends into the: tunnel 1, network 0
Dropped due to: excess delay 0, tmit queue full 0
Max on any queue: num packets: 0, num bytes: 0
Max outstanding: 0

```

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]**

DETAILED STEPS

Step 1

enable

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Router> enable
```

Step 2

show interfaces tunnel *number* [accounting]

Use this command to show that traffic is being transmitted through the RBSCP tunnel.

Example:

```
Router# show interfaces tunnel 0
Tunnel0 is up, line protocol is down
Hardware is Tunnel
Internet address is 172.17.1.4/24
MTU 1514 bytes, BW 9 Kbit, DLY 500000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
Encapsulation TUNNEL, loopback not set
Keepalive not set
Tunnel source 172.17.1.2, destination 172.20.1.3
Tunnel protocol/transport RBSCP/IP, key disabled, sequencing disabled
Tunnel TTL 255
Checksumming of packets disabled
Tunnel transmit bandwidth 1000 (kbps)
Tunnel receive bandwidth 8000 (kbps)
RBSCP operational state: invalid (0h)
RBSCP operating mode: (2EEh) delay dual-delay drop-long-delay ack-split window-t
window step: 3
drop scale : 0
ACK split size: 6
input drop scale: 5
initial TSN: 1h
fuzz factor: 0
next TSN: 1h
next sequence: 1h
current outstanding: 0
max out per RTT: 550000
packets since SACK: 0
cumulative ack: 0h
TSN at SACK: 1h
last cumulative ack: 0h
last delivered TSN: 0h
next FWDTSN corr: 0h
RTO: 704 ms
RTT: 550 ms      srtt_sa: 0      srtt_sv: 4
sentQ: num packets: 0, num bytes: 0
tmitQ: num packets: 0, num bytes: 0
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/0 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 packets output, 0 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets
0 output buffer failures, 0 output buffers swapped out
```

Configuration Examples for Implementing Tunnels

- [Example Configuring GRE IPv4 Tunnels, page 288](#)
- [Example: Configuring GRE IPv6 Tunnels, page 289](#)
- [Example Configuring GRE Tunnel IP Source and Destination VRF Membership, page 290](#)
- [Example Routing Two AppleTalk Networks Across an IP-Only Backbone, page 291](#)
- [Example Routing a Private IP Network and a Novell Network Across a Public Service Provider, page 292](#)
- [Example Configuring a CTunnel, page 293](#)
- [Example Configuring GRE CLNS CTunnels to Carry IPv4 and IPv6 Packets, page 294](#)
- [Example Configuring Manual IPv6 Tunnels, page 295](#)
- [Example Configuring 6to4 Tunnels, page 295](#)
- [Example Configuring IPv4-Compatible IPv6 Tunnels, page 296](#)
- [Example Configuring ISATAP Tunnels, page 296](#)
- [Example Configuring the RBSCP Tunnel, page 297](#)
- [Example Configuring Routing for the RBSCP Tunnel, page 297](#)
- [Example Configuring QoS Options on Tunnel Interfaces, page 299](#)
- [Example Configuring EoMPLS over GRE, page 300](#)

Example Configuring GRE IPv4 Tunnels

The following example shows a simple configuration of GRE tunneling. Note that Ethernet interface 0/1 is the tunnel source for Router A and the tunnel destination for Router B. Fast Ethernet interface 0/1 is the tunnel source for Router B and the tunnel destination for Router A.

Router A

```
interface Tunnel0
 ip address 10.1.1.2 255.255.255.0
 tunnel source Ethernet0/1
 tunnel destination 192.168.3.2
 tunnel mode gre ip
!
interface Ethernet0/1
 ip address 192.168.4.2 255.255.255.0
```

Router B

```
interface Tunnel0
 ip address 10.1.1.1 255.255.255.0
 tunnel source FastEthernet0/1
 tunnel destination 192.168.4.2
 tunnel mode gre ip
!
interface FastEthernet0/1
 ip address 192.168.3.2 255.255.255.0
```

The following example configures a GRE tunnel running both IS-IS and IPv6 traffic between Router A and Router B.

Router A

```

ipv6 unicast-routing
clns routing
!
interface Tunnel0
 no ip address
 ipv6 address 2001:0DB8:1111:2222::1/64
 ipv6 router isis
 tunnel source Ethernet0/0
 tunnel destination 10.0.0.2
 tunnel mode gre ip
!
interface Ethernet0/0
 ip address 10.0.0.1 255.255.255.0
!
router isis
 network 49.0000.0000.000a.00

```

Router B

```

ipv6 unicast-routing
clns routing
!
interface Tunnel0
 no ip address
 ipv6 address 2001:0DB8:1111:2222::2/64
 ipv6 router isis
 tunnel source Ethernet0/0
 tunnel destination 10.0.0.1
 tunnel mode gre ip
!
interface Ethernet0/0
 ip address 10.0.0.2 255.255.255.0
!
router isis
 network 49.0000.0000.000b.00
 address-family ipv6
 redistribute static
 exit-address-family

```

Example: Configuring GRE IPv6 Tunnels

The following example shows how to configure a GRE tunnel over an IPv6 transport. In this example, Ethernet0/0 has an IPv6 address, and this is the source address used by the tunnel interface. The destination IPv6 address of the tunnel is specified directly. In this example, the tunnel carries both IPv4 and IS-IS traffic.

```

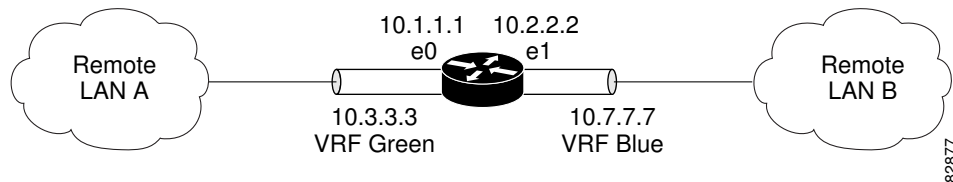
interface Tunnel0
 ip address 10.1.1.1 255.255.255.0
 ip router isis
 tunnel source Ethernet0/0
 tunnel destination 2001:DB8:1111:2222::1
 tunnel mode gre ipv6
!
interface Ethernet0/0
 no ip address
 ipv6 address 2001:DB8:1111:1111::1/64
!
router isis
 net 49.0001.0000.0000.000a.00

```

Example Configuring GRE Tunnel IP Source and Destination VRF Membership

In this example, packets received on interface e0 using VRF green, will be forwarded out of the tunnel through interface e1 using VRF blue. The figure below shows a simple tunnel scenario:

Figure 27 GRE Tunnel Diagram



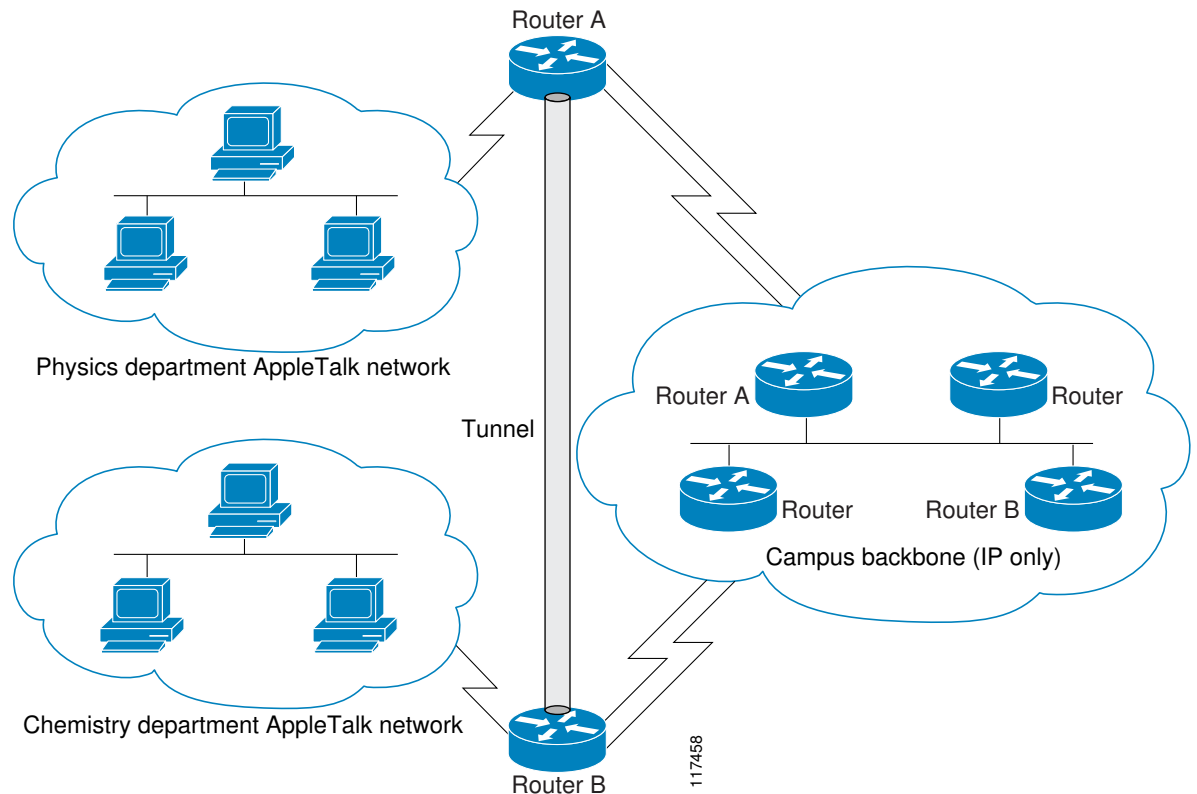
The following example shows the configuration for the tunnel in the figure above.

```
ip vrf blue
 rd 1:1
ip vrf green
 rd 1:2
interface loopback0
 ip vrf forwarding blue
 ip address 10.7.7.7 255.255.255.255
interface tunnel0
 ip vrf forwarding green
 ip address 10.3.3.3 255.255.255.0
 tunnel source loopback 0
 tunnel destination 10.5.5.5
 tunnel vrf blue
interface ethernet0
 ip vrf forwarding green
 ip address 10.1.1.1 255.255.255.0
interface ethernet1
 ip vrf forwarding blue
 ip address 10.2.2.2 255.255.255.0
 ip route vrf blue 10.5.5.5 255.255.255.0 ethernet 1
```

Example Routing Two AppleTalk Networks Across an IP-Only Backbone

The figure below is an example of connecting multiprotocol subnetworks across a single-protocol backbone. The configurations of Router A and Router B follow the figure.

Figure 28 Connecting AppleTalk Networks Across an IP-Only Backbone



Router A

```
interface ethernet 0
  description physics department AppleTalk LAN
  appletalk cable-range 4001-4001 32
  !
interface fddi 0
  description connection to campus backbone
  ip address 10.0.8.108 255.255.255.0
interface tunnel 0
  tunnel source fddi 0
  tunnel destination 10.0.21.20
  appletalk cable-range 5313-5313 1
```

Router B

```
interface ethernet 0
  description chemistry department AppleTalk LAN
  appletalk cable-range 9458-9458 3
  !
interface fddi 0
  description connection to campus backbone
```

```

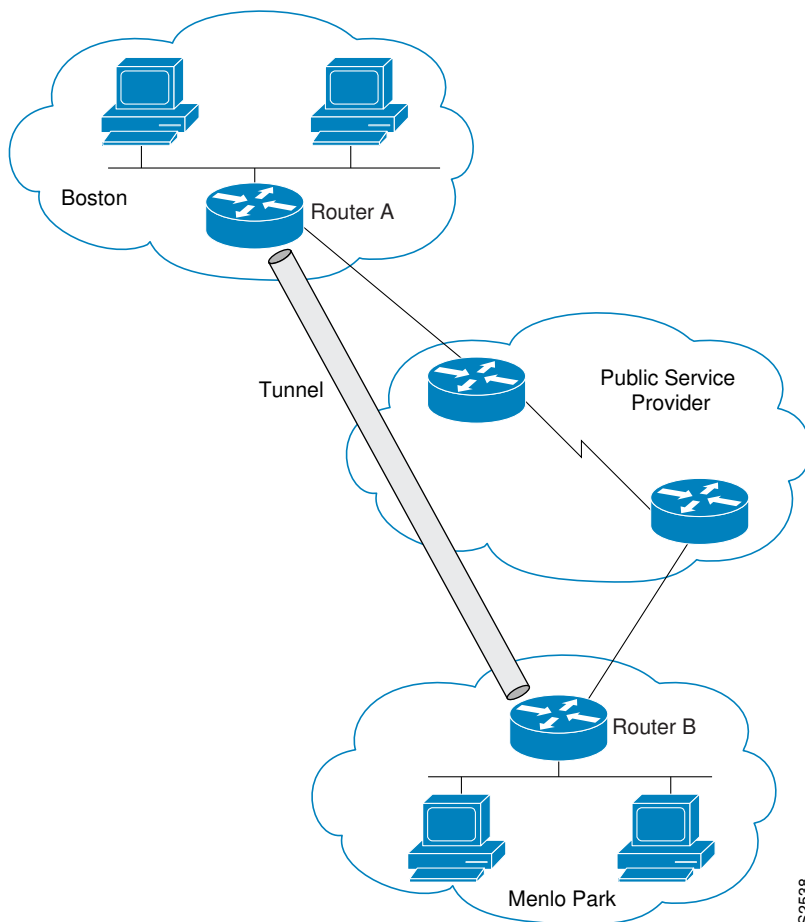
ip address 10.0.21.20 255.255.255.0
interface tunnel 0
 tunnel source fddi 0
 tunnel destination 10.0.8.108
 appletalk cable-range 5313-5313 2

```

Example Routing a Private IP Network and a Novell Network Across a Public Service Provider

The figure below is an example of routing a private IP network and a Novell network across a public service provider. The configurations of Router A and Router B follow the figure.

Figure 29 *Creating Virtual Private Networks Across WANs*



Router A

```

interface ethernet 0
 description Boston office
 ip address 10.1.1.1 255.255.255.0
 novell network 1e
 !
interface serial 0
 description connection to public service provider
 ip address 172.17.2.1 255.255.255.0
 !

```

S2538

```
interface tunnel 0
 tunnel source serial 0
 tunnel destination 172.28.5.2
 ip address 10.1.2.1 255.255.255.0
 novell network 1f
```

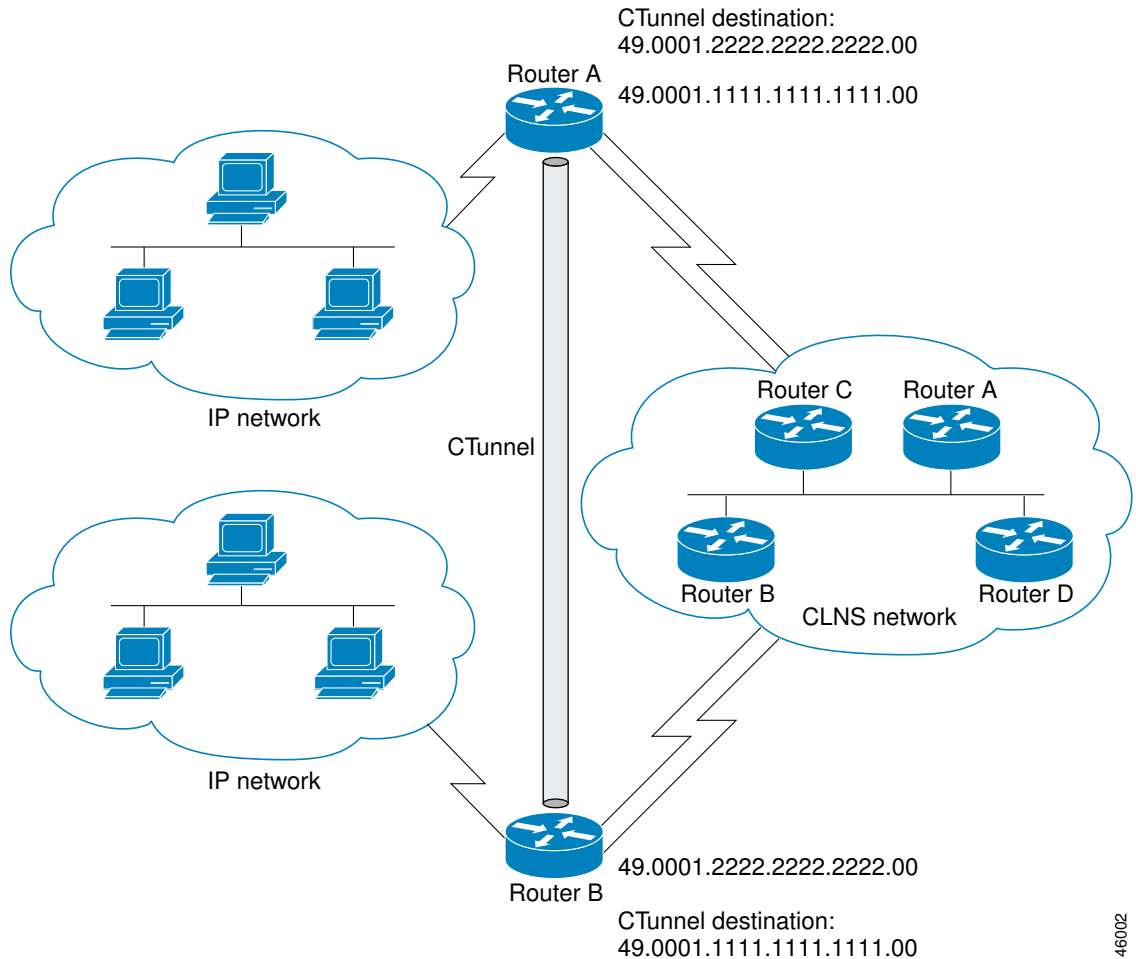
Router B

```
interface ethernet 0
 description Menlo Park office
 ip address 10.1.3.1 255.255.255.0
 novell network 31
 !
interface serial 4
 description connection to public service provider
 ip address 172.28.5.2 255.255.255.0
 !
interface tunnel 0
 tunnel source serial 4
 tunnel destination 172.17.2.1
 ip address 10.1.2.2 255.255.255.0
 novell network 1f
```

Example Configuring a CTunnel

The figure below illustrates the creation of a CTunnel between Router A and Router B, as accomplished in the configuration examples that follow.

Figure 30 *Creation of a CTunnel*



46002

Router A

```

ip routing
clns routing
interface ctunnel 102
 ip address 10.0.0.1 255.255.255.0
 ctunnel destination 49.0001.2222.2222.2222.00
interface Ethernet0/1
 clns router isis
router isis
 network 49.0001.1111.1111.1111.00
router rip
 network 10.0.0.0

```

Router B

```

ip routing
clns routing
interface ctunnel 201
 ip address 10.0.0.2 255.255.255.0
 ctunnel destination 49.0001.1111.1111.1111.00
interface Ethernet0/1
 clns router isis
router isis
 network 49.0001.2222.2222.2222.00
router rip
 network 10.0.0.0

```

Example Configuring GRE CLNS CTunnels to Carry IPv4 and IPv6 Packets

The following example configures a GRE CTunnel running both IS-IS and IPv6 traffic between Router A and Router B in a CLNS network. The **ctunnel mode gre** command provides a method of tunneling that is compliant with RFC 3147 and should allow tunneling between Cisco equipment and third-party networking devices.

Router A

```

ipv6 unicast-routing
clns routing
interface ctunnel 102
 ipv6 address 2001:0DB8:1111:2222::1/64
 ctunnel destination 49.0001.2222.2222.2222.00
 ctunnel mode gre
interface Ethernet0/1
 clns router isis
router isis
 network 49.0001.1111.1111.1111.00

```

Router B

```

ipv6 unicast-routing
clns routing
interface ctunnel 201
 ipv6 address 2001:0DB8:1111:2222::2/64
 ctunnel destination 49.0001.1111.1111.1111.00
 ctunnel mode gre
interface Ethernet0/1
 clns router isis
router isis
 network 49.0001.2222.2222.2222.00

```

To turn off GRE mode and restore the CTunnel to the default Cisco encapsulation routing only between endpoints on Cisco equipment, use either the **no ctunnel mode** command or the **ctunnel mode cisco** command. The following example shows the same configuration modified to transport only IPv4 traffic.

Router A

```
ip routing
clns routing
interface ctunnel 102
 ip address 10.2.2.5 255.255.255.0
 ctunnel destination 49.0001.2222.2222.2222.00
 ctunnel mode cisco
interface Ethernet0/1
 clns router isis
router isis
 network 49.0001.1111.1111.1111.00
```

Router B

```
ip routing
clns routing
interface ctunnel 201
 ip address 10.0.0.5 255.255.255.0
 ctunnel destination 49.0001.1111.1111.1111.00
 ctunnel mode cisco
interface Ethernet0/1
 clns router isis
router isis
 network 49.0001.2222.2222.2222.00
```

Example Configuring Manual IPv6 Tunnels

The following example configures a manual IPv6 tunnel between Router A and Router B. In the example, tunnel interface 0 for both Router A and Router B is manually configured with a global IPv6 address. The tunnel source and destination addresses are also manually configured.

Router A

```
interface ethernet 0
 ip address 192.168.99.1 255.255.255.0
interface tunnel 0
 ipv6 address 2001:0db8:c18:1::3/126
 tunnel source ethernet 0
 tunnel destination 192.168.30.1
 tunnel mode ipv6ip
```

Router B

```
interface ethernet 0
 ip address 192.168.30.1 255.255.255.0
interface tunnel 0
 ipv6 address 2001:0db8:c18:1::2/126
 tunnel source ethernet 0
 tunnel destination 192.168.99.1
 tunnel mode ipv6ip
```

Example Configuring 6to4 Tunnels

The following example configures a 6to4 tunnel on a border router in an isolated IPv6 network. The IPv4 address is 192.168.99.1, which translates to the IPv6 prefix of 2002:c0a8:6301::/48. The IPv6 prefix is subnetted into 2002:c0a8:6301::/64 for the tunnel interface: 2002:c0a8:6301:1::/64 for the first IPv6 network and 2002:c0a8:6301:2::/64 for the second IPv6 network. The static route ensures that any other traffic for the IPv6 prefix 2002::/16 is directed to tunnel interface 0 for automatic tunneling.

```
interface Ethernet0
```

```

description IPv4 uplink
ip address 192.168.99.1 255.255.255.0
!
interface Ethernet1
description IPv6 local network 1
ipv6 address 2002:c0a8:6301:1::1/64
!
interface Ethernet2
description IPv6 local network 2
ipv6 address 2002:c0a8:6301:2::1/64
!
interface Tunnel0
description IPv6 uplink
no ip address
ipv6 address 2002:c0a8:6301::1/64
tunnel source Ethernet0
tunnel mode ipv6ip 6to4
!
ipv6 route 2002::/16 Tunnel0

```

Example Configuring IPv4-Compatible IPv6 Tunnels

The following example configures an IPv4-compatible IPv6 tunnel that allows BGP to run between a number of routers without having to configure a mesh of manual tunnels. Each router has a single IPv4-compatible tunnel, and multiple BGP sessions can run over each tunnel, one to each neighbor. Ethernet interface 0 is used as the tunnel source. The tunnel destination is automatically determined by the IPv4 address in the low-order 32 bits of an IPv4-compatible IPv6 address. Specifically, the IPv6 prefix 0:0:0:0:0:0 is concatenated to an IPv4 address (in the format 0:0:0:0:0:0:A.B.C.D or ::A.B.C.D) to create the IPv4-compatible IPv6 address. Ethernet interface 0 is configured with a global IPv6 address and an IPv4 address (the interface supports both the IPv6 and IPv4 protocol stacks).

Multiprotocol BGP is used in the example to exchange IPv6 reachability information with the peer 10.67.0.2. The IPv4 address of Ethernet interface 0 is used in the low-order 32 bits of an IPv4-compatible IPv6 address and is also used as the next-hop attribute. Using an IPv4-compatible IPv6 address for the BGP neighbor allows the IPv6 BGP session to be automatically transported over an IPv4-compatible tunnel.

```

interface tunnel 0
tunnel source Ethernet 0
tunnel mode ipv6ip auto-tunnel
interface ethernet 0
ip address 10.27.0.1 255.255.255.0
ipv6 address 3000:2222::1/64
router bgp 65000
no synchronization
no bgp default ipv4-unicast
neighbor ::10.67.0.2 remote-as 65002
address-family ipv6
neighbor ::10.67.0.2 activate
neighbor ::10.67.0.2 next-hop-self
network 2001:2222:d00d:b10b::/64

```

Example Configuring ISATAP Tunnels

The following example shows the tunnel source defined on Ethernet 0 and the **tunnel mode** command used to configure the ISATAP tunnel. Router advertisements are enabled to allow client autoconfiguration.

```

interface Tunnell
tunnel source ethernet 0
tunnel mode ipv6ip isatap
ipv6 address 2001:0DB8::/64 eui-64
no ipv6 nd suppress-ra

```

Example Configuring the RBSCP Tunnel

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
```

Example Configuring Routing for the RBSCP Tunnel

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

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

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 Tunnel1
 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

```

Example Configuring QoS Options on Tunnel Interfaces

The following sample configuration applies generic traffic shaping (GTS) directly on the tunnel interface. In this example the configuration shapes the tunnel interface to an overall output rate of 500 kbps. For more details on GTS, see the "Regulating Packet Flow Using Traffic Shaping" chapter of the Cisco IOS Quality of Service Solutions Configuration Guide.

```
interface Tunnel0
 ip address 10.1.2.1 255.255.255.0
 traffic-shape rate 500000 125000 125000 1000
 tunnel source 10.1.1.1
 tunnel destination 10.2.2.2
```

The following sample configuration shows how to apply the same shaping policy to the tunnel interface with the Modular QoS CLI (MQC) commands. For more details on MQC, see the "Modular Quality of Service Command-Line Interface" chapter of the Cisco IOS Quality of Service Solutions Configuration Guide .

```
policy-map tunnel
 class class-default
  shape average 500000 125000 125000
!
interface Tunnel0
 ip address 10.1.2.1 255.255.255.0
 service-policy output tunnel
 tunnel source 10.1.35.1
 tunnel destination 10.1.35.2
```

Policing Example

When an interface becomes congested and packets start to queue, you can apply a queueing method to packets that are waiting to be transmitted. Cisco IOS logical interfaces--tunnel interfaces in this example--do not inherently support a state of congestion and do not support the direct application of a service policy that applies a queueing method. Instead, you need to apply a hierarchical policy. Create a "child" or lower-level policy that configures a queueing mechanism, such as low latency queueing with the **priority** command and class-based weighted fair queueing (CBWFQ) with the **bandwidth** command.

```
policy-map child
 class voice
  priority 512
```

Create a "parent" or top-level policy that applies class-based shaping. Apply the child policy as a command under the parent policy because admission control for the child class is done according to the shaping rate for the parent class.

```
policy-map tunnel
 class class-default
  shape average 2000000
  service-policy child
```

Apply the parent policy to the tunnel interface.

```
interface tunnel0
 service-policy tunnel
```

In the following example, a tunnel interface is configured with a service policy that applies queueing without shaping. A log message is displayed noting that this configuration is not supported.

```
interface tunnell
 service-policy output child
 Class Based Weighted Fair Queueing not supported on this interface
```

For more details on QoS traffic policing, see the Cisco IOS Quality of Service Solutions Configuration Guide .

Example Configuring EoMPLS over GRE

The following sample provides the EoMPLS over GRE configuration sequence:

PE1 Configuration

```
vrf definition VPN1
 rd 100:1
 address-family ipv4
 route-target both 100:1
 exit-address-family
 !
 mpls label protocol ldp
 mpls ldp neighbor 10.10.10.11 targeted
 mpls ldp router-id Loopback0 force
 !
 interface Tunnel0
 ip address 100.1.1.11 255.255.255.0
 mpls label protocol ldp
 mpls ip
 keepalive 10 3
 tunnel source TenGigabitEthernet2/1/0
 tunnel destination 50.1.3.2
 !
 interface Loopback0
 ip address 10.10.10.10 255.255.255.255
 !
 interface TenGigabitEthernet2/1/0
 mtu 9216
 ip address 10.1.1.1 255.255.255.0
 !
 interface TenGigabitEthernet9/1
 no ip address
 !
 interface TenGigabitEthernet9/1.11
 vrf forwarding VPN1
 encapsulation dot1q 300
 ip address 192.168.1.1 255.255.255.0
 !
 interface TenGigabitEthernet9/2
 mtu 9216
 no ip address
 xconnect 10.10.10.11 200 encapsulation mpls
 !
 router bgp 65000
 bgp log-neighbor-changes
 neighbor 10.10.10.11 remote-as 65000
 neighbor 10.10.10.11 update-source Loopback0
 neighbor 192.168.1.2 remote-as 100
 !
 address-family vpnv4
 neighbor 10.10.10.11 activate
 neighbor 10.10.10.11 send-community extended
 !
 address-family ipv4 vrf VPN1
 no synchronization
 neighbor 192.168.1.2 remote-as 100
 neighbor 192.168.1.2 activate
 neighbor 192.168.1.2 send-community extended
 !
 ip route 10.10.10.11 255.255.255.255 Tunnel0
 ip route 10.1.3.0 255.255.255.0 10.1.1.2
PE2 Configuration
vrf definition VPN1
 rd 100:1
 address-family ipv4
```

```

route-target both 100:1
exit-address-family
!
mpls ldp neighbor 10.10.10.10 targeted
mpls label protocol ldp
mpls ldp router-id Loopback0 force
!
interface Tunnel0
 ip address 100.1.1.10 255.255.255.0
 mpls label protocol ldp
 mpls ip
 keepalive 10 3
 tunnel source TenGigabitEthernet3/3/0
 tunnel destination 10.1.1.1
!
interface Loopback0
 ip address 10.10.10.11 255.255.255.255
!
interface TenGigabitEthernet2/1
 mtu 9216
 no ip address
 xconnect 10.10.10.10 200 encapsulation mpls
!
interface TenGigabitEthernet2/3
 mtu 9216
 no ip address
!
interface TenGigabitEthernet2/3.11
 vrf forwarding VPN1
 encapsulation dot1Q 300
 ip address 192.168.2.1 255.255.255.0
!
interface TenGigabitEthernet3/3/0
 mtu 9216
 ip address 10.1.3.2 255.255.255.0
!
router bgp 65000
 bgp log-neighbor-changes
 neighbor 10.10.10.10 remote-as 65000
 neighbor 10.10.10.10 update-source Loopback0
 neighbor 192.168.2.2 remote-as 200
!
 address-family vpnv4
  neighbor 10.10.10.10 activate
  neighbor 10.10.10.10 send-community extended
 exit-address-family
!
 address-family ipv4 vrf VPN1
  no synchronization
  neighbor 192.168.2.2 remote-as 200
  neighbor 192.168.2.2 activate
  neighbor 192.168.2.2 send-community extended
 exit-address-family
!
ip route 10.10.10.10 255.255.255.255 Tunnel0
ip route 10.1.1.0 255.255.255.0 10.1.3.1

```

Where to Go Next

If you have implemented IPv6 tunnels, you may want to proceed to one of the following modules:

- If you have configured an automatic 6to4 tunnel, you can design your IPv6 network around the /48 6to4 prefix that you have created from your IPv4 address.
- If you want to implement routing protocols, see the "Implementing RIP for IPv6," "Implementing IS-IS for IPv6," "Implementing OSPF for IPv6," or "Implementing Multiprotocol BGP for IPv6" modules.
- If you want to implement security features for your IPv6 network, see the "Implementing Security for IPv6" module.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Tunnel commands: complete command syntax, command mode, defaults, command history, usage guidelines, and examples	<i>Cisco IOS Interface and Hardware Component Command Reference</i>
CLNS commands: complete command syntax, command mode, defaults, command history, usage guidelines, and examples	<i>Cisco IOS ISO CLNS Command Reference</i>
IP commands: complete command syntax, command mode, defaults, command history, usage guidelines, and examples	<ul style="list-style-type: none"> • <i>Cisco IOS IP Addressing Services Command Reference</i> • <i>Cisco IOS IP Application Services Command Reference</i> • <i>Cisco IOS IP Routing Protocols Command Reference</i>
IPv6 commands: complete command syntax, command mode, defaults, command history, usage guidelines, and examples	<i>Cisco IOS IPv6 Command Reference</i>
IPv6 Features	<i>Cisco IOS IPv6 Configuration Guide</i>
Regulating Packet Flow on a Per-Interface Basis - Using Generic Traffic Shaping	<i>Cisco IOS Quality of Service Solutions Configuration Guide</i>
Modular QoS CLI configuration	<i>Cisco IOS Quality of Service Solutions Configuration Guide</i>
Virtual interface configuration	<i>Cisco IOS Interface and Hardware Component Configuration Guide</i>
Configuration example for GRE over IP Security (IPSec) where the GRE/IPSec tunnel is going through a firewall doing Network Address Translation (NAT)	Configuring IPSec/GRE with NAT
Configuring Multiprotocol Label Switching	<i>Cisco IOS Multiprotocol Label Switching Configuration Guide</i>

Standards

Standard	Title
No new or modified standards are supported, and support for existing standards has not been modified.	--

MIBs

MIB	MIBs Link
No new or modified MIBs are supported, and support for existing MIBs has not been modified.	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

RFC	Title
RFC 791	<i>Internet Protocol</i>
RFC 1191	<i>Path MTU Discovery</i>
RFC 1323	<i>TCP Extensions for High Performance</i>
RFC 1483	<i>Multiprotocol Encapsulation over ATM Adaptation Layer 5</i>
RFC 2003	<i>IP Encapsulation Within IP</i>
RFC 2018	<i>TCP Selective Acknowledgment Options</i>
RFC 2460	<i>Internet Protocol, Version 6 (IPv6)</i>
RFC 2473	<i>Generic Packet Tunneling in IPv6 Specification</i>
RFC 2474	<i>Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers</i>
RFC 2516	<i>A Method for Transmitting PPP over Ethernet (PPPoE)</i>
RFC 2547	BGP/MPLS VPNs
RFC 2780	IANA Allocation Guidelines for Values in the Internet Protocol and Related Headers
RFC 2784	Generic Routing Encapsulation (GRE)
RFC 2890	Key and Sequence Number Extensions to GRE
RFC 2893	Transition Mechanisms for IPv6 Hosts and Routers

RFC	Title
RFC 3056	Connection of IPv6 Domains via IPv4 Clouds
RFC 3147	Generic Routing Encapsulation over CLNS Networks

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Implementing Tunnels

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 21 **Feature Information for Implementing Tunnels**

Feature Name	Releases	Feature Configuration Information
CEF-Switched Multipoint GRE Tunnels	12.2(8)T 15.0(1)M	<p>The CEF-Switched Multipoint GRE Tunnels feature enables CEF switching of IP traffic to and from multipoint GRE tunnels. Tunnel traffic can be forwarded to a prefix through a tunnel destination when both the prefix and the tunnel destination are specified by the application.</p> <p>This feature introduces CEF switching over multipoint GRE tunnels. Previously, only process switching was available for multipoint GRE tunnels.</p> <p>No commands were introduced or modified by this feature.</p>
CLNS Support for GRE Tunneling of IPv4 and IPv6 Packets in CLNS Networks	12.3(7)T 12.2(25)S 12.2(33)SRA	<p>Support of the GRE tunnel mode allows Cisco CTunnels to transport IPv4 and IPv6 packets over CLNS-only networks in a manner that allows interoperation between Cisco networking equipment and that of other vendors. This feature provides compliance with RFC 3147.</p> <p>The following command was introduced by this feature: ctunnel mode.</p>
GRE Tunnel IP Source and Destination VRF Membership	12.0(23)S 12.2(20)S 12.2(27)SBC 12.3(2)T 12.2(33)SRA 12.2(33)SRB 12.2(31)SB5 12.4(15)T	<p>Allows you to configure the source and destination of a tunnel to belong to any VPN VRF table.</p> <p>In 12.0(23)S, this feature was introduced.</p> <p>In 12.2(20)S this feature became available on Cisco 7304 router using the NSE-100 in the PXF processing path.</p> <p>In 12.2(31)SB5, support was added for the Cisco 10000 series router for the PRE2 and PRE3.</p> <p>The following command was introduced to support this feature: tunnel vrf.</p>

Feature Name	Releases	Feature Configuration Information
GRE Tunnel Keepalive	12.2(8)T 12.0(23)S 15.0(1)M Cisco IOS XE 3.1.0SG 15.3(1)S	<p>The GRE Tunnel Keepalive feature provides the capability of configuring keepalive packets to be sent over IP-encapsulated generic routing encapsulation (GRE) tunnels. You can specify the rate at which keepalives will be sent and the number of times that a device will continue to send keepalive packets without a response before the interface becomes inactive. GRE keepalive packets may be sent from both sides of a tunnel or from just one side.</p> <p>The following command was introduced by this feature: keepalive (tunnel interfaces).</p>
IP Tunnel-- SSO	15.1(1)SY	<p>High availability support was added to IP Tunnels.</p> <p>No new commands were introduced or modified by this feature.</p>
Rate-Based Satellite Control Protocol	12.3(7)T	<p>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.</p> <p>The following commands were introduced or modified by this feature: 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.</p>

Feature Name	Releases	Feature Configuration Information
Tunnel ToS	12.0(17)S 12.0(17)ST 12.2(8)T 12.2(14)S 15.0(1)M	<p>The Tunnel ToS feature allows you to configure the ToS and Time-to-Live (TTL) byte values in the encapsulating IP header of tunnel packets for an IP tunnel interface on a router. The Tunnel ToS feature is supported on Cisco Express Forwarding (CEF), fast switching, and process switching forwarding modes.</p> <p>The following commands were introduced or modified by this feature: show interfaces tunnel, tunnel tos, tunnel ttl.</p>

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



Tunnel Route Selection

The Tunnel Route Selection feature allows the tunnel transport to be routed using a subset of the routing table. When there are equal-cost routes to a tunnel destination, normal tunnel transport behavior is to use one of the available routes chosen at random. The Tunnel Route Selection feature allows the explicit configuration of the outgoing interface for the tunnel transport.

- [Finding Feature Information, page 309](#)
- [Prerequisites for Tunnel Route Selection, page 309](#)
- [Restrictions for Tunnel Route Selection, page 309](#)
- [Information About Tunnel Route Selection, page 310](#)
- [How to Configure Tunnel Route Selection, page 311](#)
- [Configuration Examples for Tunnel Route Selection, page 312](#)
- [Additional References, page 313](#)
- [Feature Information for Tunnel Route Selection, page 314](#)

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.

Prerequisites for Tunnel Route Selection

Tunnel interfaces are configured.

Restrictions for Tunnel Route Selection

This feature is supported in the following tunnel modes only:

- Generic Routing Encapsulation (GRE) IP
- GRE Multipoint
- IP in IP
- Mobile User Datagram Protocol (UDP)

This feature is not supported on a tunnel when the tunnel transport is a GRE Multipoint tunnel.

Supported Configuration

```
interface tunnel 0
  tunnel mode gre multipoint
  tunnel route-via tunnel 1
interface tunnel 1
  tunnel mode gre ip
```

Unsupported Configuration

```
interface tunnel 0
  tunnel mode gre multipoint
  tunnel route-via tunnel 1
interface tunnel 1
  tunnel mode gre multipoint
```

Information About Tunnel Route Selection

- [Tunnel Transport Behavior, page 310](#)

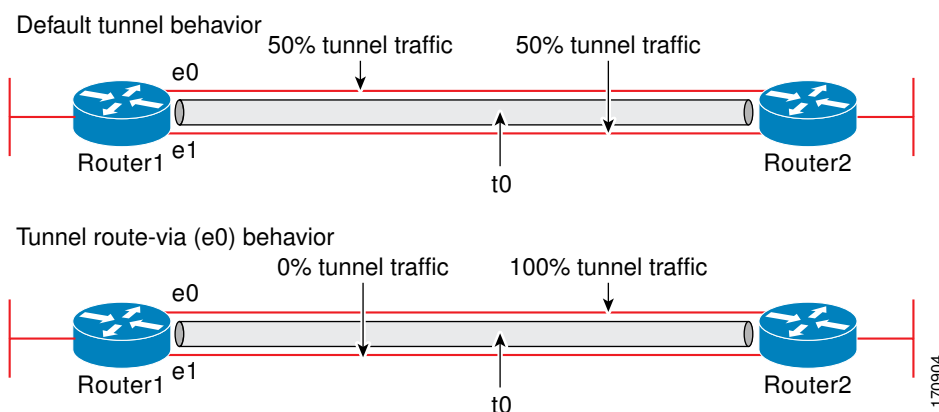
Tunnel Transport Behavior

The Tunnel Route Selection feature allows the tunnel transport to be routed using a subset of the routing table by specifying the outgoing interface of the tunnel transport.

The Tunnel Route Selection feature is not the same as an implementation of policy-based routing for the tunnel transport. The Tunnel Route Selection feature will forward traffic using only a subset of the route table, and it cannot introduce routing loops into the network.

The figure below compares default tunnel behavior with the Tunnel Route Selection behavior.

Figure 31 Tunnel Route Selection Traffic



How to Configure Tunnel Route Selection

- [Configuring Tunnel Route Selection, page 311](#)

Configuring Tunnel Route Selection

Perform the following steps to specify the outgoing interface of the tunnel transport to route the tunnel transport using a subset of the routing table.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *interface-number*
4. **tunnel route-via** *interface-type interface-number* {**mandatory** | **preferred**}
5. **end**

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	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	interface tunnel <i>interface-number</i> Example: Router(config)# interface tunnel 0	Configures a tunnel interface and enters interface configuration mode.
Step 4	tunnel route-via <i>interface-type interface-number</i> { mandatory preferred } Example: Router(config-if)# tunnel route-via ethernet0 mandatory	Specifies the outgoing interface to be used by the tunnel transport.

Command or Action	Purpose
Step 5 end Example: Router(config-if)# end	Returns to privileged EXEC mode.

- [Troubleshooting Tips, page 312](#)
- [What to Do Next, page 312](#)

Troubleshooting Tips

To troubleshoot your configuration, use the **debug tunnel route-via** command in privileged EXEC mode. The following is sample output from the **debug tunnel route-via** command after the **tunnel route-via** command was used to route the tunnel transport explicitly using a subset of the routing table.

```
Router# debug tunnel route-via
Tunnel route-via debugging is on
Router#
*May 23 08:40:53.707: TUN-VIA: Tunnel0 candidate route-via Ethernet0/0, next hop 10.73.2.1
*May 23 08:40:53.707: TUN-VIA: Tunnel0 route-via action is forward
*May 23 08:41:03.719: TUN-VIA: Tunnel0 candidate route-via Ethernet0/0, next hop 10.73.2.1
*May 23 08:41:03.719: TUN-VIA: Tunnel0 route-via action is forward
Router# undebug tunnel route-via
Tunnel route-via debugging is off
```

What to Do Next

You can verify the tunnel route selection configuration. To verify your configuration, use the **show interfaces tunnel** command in privileged EXEC mode. The following example shows that the tunnel transport is routed using a subset of the routing table by specifying the outgoing interface of the tunnel transport.

```
Router# show running-config interface tunnel 0
Building configuration...
Current configuration : 147 bytes
!
interface Tunnel0
 ip unnumbered Loopback0
 tunnel source Loopback0
 tunnel destination 10.73.0.102
 tunnel route-via Ethernet0 preferred
end
Router# show interfaces tunnel 0 | include route-via
Tunnel route-via feature is on [Ethernet0, preferred]
```

Configuration Examples for Tunnel Route Selection

- [Example Configuring Tunnel Route Selection, page 313](#)

Example Configuring Tunnel Route Selection

The following example shows Tunnel 0 configured to use Ethernet interface 0 as its preferred outgoing transport interface. Traffic that exits the router using the tunnel 0 interface will be sent out of Ethernet interface 0 if there is a route to the tunnel destination out of Ethernet interface 0. If there is no route out of Ethernet interface 0, the traffic will be forwarded as if the Tunnel Route Selection feature were not configured.

If the **tunnel route-via** *interface-type interface-number* **mandatory** command is configured, and there is no route to the tunnel destination using that interface, a point-to-point tunnel interface will go into a down state.

```
Router> enable
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface tunnel 0
Router(config-if)# tunnel route-via ethernet0 preferred
Router(config-if)# end
Router# show running-config interface tunnel 0
Building configuration...
Current configuration : 147 bytes
!
interface Tunnel0
 ip unnumbered Loopback0
 tunnel source Loopback0
 tunnel destination 10.73.0.102
 tunnel route-via Ethernet0 preferred
end
```

Additional References

Related Documents

Related Topic	Document Title
Implementing tunnels	<i>Cisco IOS Interface and Hardware Component Configuration Guide</i>
Interface and hardware component commands: tunnel route-via, show interfaces tunnel	<i>Cisco IOS Interface and Hardware Component Command Reference</i>
Debug command: debug tunnel route-via	<i>Cisco IOS Debug Command Reference</i>

Standards

Standard	Title
None	--

MIBs

MIB	MIBs Link
None	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

RFC	Title
None	--

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Tunnel Route Selection

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 22 **Feature Information for Tunnel Route Selection**

Feature Name	Releases	Feature Information
Tunnel Route Selection	12.4(11)T 15.0(1)M	<p>The Tunnel Route Selection feature allows the tunnel transport to be routed using a subset of the routing table. When there are equal-cost routes to a tunnel destination, normal tunnel transport behavior is to use one of the available routes chosen at random. The Tunnel Route Selection feature allows the explicit configuration of the outgoing interface for the tunnel transport.</p> <p>The following commands were introduced or modified: debug tunnel route-via, tunnel route-via, show interfaces tunnel.</p>

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



MPLS VPN over mGRE

The MPLS VPN over mGRE feature overcomes the requirement that a carrier support multiprotocol label switching (MPLS) by allowing you to provide MPLS connectivity between networks that are connected by IP-only networks. This allows MPLS label switched paths (LSPs) to use generic routing encapsulation (GRE) tunnels to cross routing areas, autonomous systems, and internet service providers (ISPs). When MPLS VPNs are configured over multipoint GRE (mGRE) you can deploy layer-3 (L3) provider edge (PE) based virtual private network (VPN) services using a standards-based IP core. This allows you to provision the VPN services without using the overlay method.

You can configure mGRE tunnels to create a multipoint tunnel network that overlays an IP backbone. This overlay connects PE routers to transport VPN traffic. In addition, when MPLS VPNs are configured over mGRE you can deploy L3 PE-based VPN services using a standards-based IP core. This allows you to provision the VPN services without using the overlay method. When MPLS VPN over mGRE is configured, the system uses IPv4-based mGRE tunnels to encapsulate VPN-labeled IPv4 and IPv6 packets between PEs.

- [Finding Feature Information, page 317](#)
- [Prerequisites for MPLS VPN over mGRE, page 317](#)
- [Restrictions for MPLS VPN over mGRE, page 318](#)
- [Information About MPLS VPN over mGRE, page 318](#)
- [How to Configure MPLS VPN over mGRE, page 320](#)
- [Configuration Examples for MPLS VPN over mGRE, page 328](#)
- [Additional References, page 330](#)
- [Feature Information for MPLS VPN over mGRE, page 331](#)

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.

Prerequisites for MPLS VPN over mGRE

Before you configure MPLS VPN with mGRE tunnels, ensure that the MPLS VPN is configured and working properly. See the "Configuring MPLS Layer 3 VPNs" module for information about setting up MPLS VPNs.

Restrictions for MPLS VPN over mGRE

- MPLS VPN over mGRE is supported on the Cisco 7600 series routers using the ES-40 line card and the SIP 400 line card as core facing cards.
- Tunnelled tag traffic must enter the router through a line card that supports MPLS VPN over mGRE.
- Each PE router supports one tunnel configuration only.
- MPLS VPN over mGRE does not support the transportation of multicast traffic between VPNs.
- When a GRE tunnel has the same destination address and source address as the mGRE, the tunnel gets route-cache switched.
- The packets that require fragmentation get route cache-switched.
- When an L3VPN profile is removed and added back, then you should clear the Border Gateway Protocol (BGP) using the **clear ip bgp soft** command.
- When an mGRE tunnel is created, a dummy tunnel is also created.
- The loopback or IP address used in the update source of the BGP configuration should be the same as that of the transport source of the L3VPN profile.
- mGRE is not stateful switchover (SSO) compliant. However, both mGRE and SSO coexist.
- mGRE and multicast distribution tree (MDT) tunnel should not be configured with the same loopback address.

The limitations for MPLS VPN over mGRE feature are as follows:

- - Not all GRE options are supported in the hardware (for example, GRE extended header and GRE key).
 - Checking identical VLANs (Internet Control Message Protocol [ICMP] redirect) is not supported on the tunnels.
 - Features such as unicast reverse path forwarding (uRPF) and BGP policy accounting are not supported on the tunnels.

Information About MPLS VPN over mGRE

- [MPLS VPN over mGRE, page 318](#)

MPLS VPN over mGRE

GRE is a point-to-point tunneling protocol where two peers form the endpoints of the tunnel. It is designed to encapsulate network-layer packets inside IP tunneling packets. mGRE is a similar protocol with a single endpoint at one side of the tunnel connected to multiple endpoints at the other side of the tunnel. The mGRE tunnel provides a common link between branch offices that connect to the same VPN. Because mGRE is a point-to-multipoint model, fully meshed GRE tunnels are not required to interconnect MPLS VPN PE devices.

MPLS is a widely deployed VPN internet architecture. MPLS requires that all core routers in the network support MPLS. This feature is useful in networks where the service provider uses a backbone carrier to provide connectivity.

The MPLS VPN over mGRE feature overcomes the requirement of carrier support MPLS by allowing you to provide MPLS connectivity between networks that are connected by IP-only networks. This allows MPLS LSPs to use GRE tunnels to cross routing areas, autonomous systems, and ISPs.

When MPLS VPNs are configured over mGRE you can deploy L3 PE-based VPN services using a standards-based IP core. This allows you to provision the VPN services without using LSP or a Label Distribution Protocol (LDP). The system uses IPv4-based mGRE tunnels to encapsulate VPN-labeled IPv4 and IPv6 packets between PEs.

The MPLS VPN over mGRE feature also allows you to deploy existing MPLS VPN LSP-encapsulated technology concurrently with MPLS VPN over mGRE and enables the system to determine which encapsulation method is used to route specific traffic. The ingress PE router determines which encapsulation technology to use when a packet is sent to the remote PE router.

This section includes information on the following topics on MPLS VPN over mGRE feature:

- [Route Maps, page 319](#)
- [Tunnel Endpoint Discovery and Forwarding, page 319](#)
- [Tunnel Decapsulation, page 320](#)
- [Tunnel Source, page 320](#)
- [IPv6 VPN, page 320](#)

Route Maps

By default, VPN traffic is sent using an LSP. The MPLS VPN over mGRE feature uses user-defined route maps to determine which VPN prefixes are reachable over an mGRE tunnel and which VPN prefixes are reachable using an LSP. The route map is applied to advertisements for VPNv4 and VPNv6 address families. The route map uses a next hop tunnel table to determine the encapsulation method for the VPN traffic.

To route traffic over the mGRE tunnel, the system creates an alternative address space that shows that all next hops are reached by encapsulating the traffic in an mGRE tunnel. To configure a specific route to use an mGRE tunnel, the user adds an entry for that route to the route map. The new entry remaps the Network Layer Reachability Information (NLRI) of the route to the alternative address space. If there is no remap entry in the route map for a route, then traffic on that route is forwarded over an LSP.

When the user configures MPLS VPN over mGRE, the system automatically provisions the alternative address space, normally held in the tunnel-encapsulated virtual routing and forwarding (VRF) instance. To ensure that all traffic reachable through the address space is encapsulated in an mGRE tunnel, the system installs a single default route out of a tunnel. The system also creates a default tunnel on the route map. The user can attach this default route map to the appropriate BGP updates.

Tunnel Endpoint Discovery and Forwarding

In order for the MPLS VPN over mGRE feature to function correctly, the system must be able to discover the remote PEs in the system and construct tunnel forwarding information for these remote PEs. In addition the system must be able to detect when a remote PE is no longer valid and remove the tunnel forwarding information for that PE.

If an ingress PE receives a VPN advertisement over BGP, it uses the route target attributes (which it inserts into the VRF) and the MPLS VPN label from the advertisement, to associate the prefixes with the appropriate customer. The next hop of the inserted route is set to the NLRI of the advertisement.

The advertised prefixes contain information about remote PEs in the system (in the form of NLRIs), and the PE uses this information to notify the system when an NLRI becomes active or inactive. The system uses this notification to update the PE forwarding information.

When the system receives notification of a new remote PE, it adds the information to the tunnel endpoint database, which causes the system to create an adjacency associated with the tunnel interface. The adjacency description includes information on the encapsulation and other processing that the system must perform to send encapsulated packets to the new remote PE.

The adjacency information is placed into the tunnel encapsulated VRF. When a user remaps a VPN NLRI to a route in the VRF (using the route map), the system links the NLRI to the adjacency; therefore the VPN is linked to a tunnel.

Tunnel Decapsulation

When the egress PE receives a packet from a tunnel interface that uses the MPLS VPN over mGRE feature, the PE decapsulates the packet to create a VPN label tagged packet, and sends the packet to the MPLS forwarding (MFI) code.

Tunnel Source

The MPLS VPN over mGRE feature uses a single tunnel configured as an mGRE tunnel to configure a system with a large number of endpoints (remote PEs). To identify the origin of tunnel-encapsulated packets, the system uses the tunnel source information.

At the transmitting (ingress) PE, when a VPN packet is sent to a tunnel, the tunnel destination is the NLRI. At a receiving (egress) PE, the tunnel source is the address that the packets encapsulated in the mGRE tunnel are received on. Therefore, at the egress PE the packet destination must match the NLRI from the local PE.

IPv6 VPN

If the advertising PE router has an IPv6 address then the NLRI must also be an IPv6 address (regardless of the network between the PEs). If the network between the PEs is IPv4 based, the system creates the IPv6 address of the advertising PE using an IPv4 mapped address in the following form: ::FFFF:IPv4-PE-address. The receiving PE sets the next hop for the VPN tag IPv6 prefixes to the IPv4 address embedded in the IPv6 NLRI. This enables the PE to link VPNv6 traffic to an LSP or an mGRE tunnel in the same way it maps VPNv4 traffic.

When a PE receives VPNv6 updates, it applies the IPv6 route map. The MPLS VPN over mGRE feature uses the IPv6 route map to set the next hop information in the Tunnel_Encap VRF.

How to Configure MPLS VPN over mGRE

To deploy MPLS VPN over mGRE tunnels, you create a VRF instance, enable and configure L3 VPN encapsulation, link the route map to the application template, and set up the BGP VPNv4 and VPNv6 exchange so that updates are filtered through the route map.

- [Configuring an L3VPN Encapsulation Profile, page 320](#)
- [Configuring BGP and Route Maps, page 322](#)

Configuring an L3VPN Encapsulation Profile

This section describes how to configure an L3VPN encapsulation profile.

**Note**

Transport protocols such as IPv6, MPLS, IP, and Layer 2 Tunneling Protocol version 3 (L2TPv3) can also be used in this configuration.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **l3vpn encapsulation ip** *profile-name*
4. **transport ipv4** [**source** *interface-type interface-number*]
5. **protocol gre** [**key** *gre-key*]
6. **end**
7. **show l3vpn encapsulation ip** *profile-name*

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 configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3 l3vpn encapsulation ip <i>profile-name</i> Example: Router(config)# l3vpn encapsulation ip tunnel encap	Enters L3 VPN encapsulation configuration mode to create the tunnel.
Step 4 transport ipv4 [source <i>interface-type interface-number</i>] Example: Router(config-l3vpn-encap-ip)# transport ipv4 source loopback 0	(Optional) Specifies IPv4 transport source mode and defines the transport source interface. <ul style="list-style-type: none"> • If you use the transport ipv4 source <i>interface-type interface-number</i> command, make sure that the specified source address is used as the next hop in BGP updates advertised by the PE. • If you do not use this command, the bgp update sourceor bgp next-hop command is automatically used as the tunnel source.

Command or Action	Purpose
<p>Step 5 <code>protocol gre [key <i>gre-key</i>]</code></p> <p>Example:</p> <pre>Router(config-l3vpn-encap-ip)# protocol gre key 1234</pre>	Specifies GRE as the tunnel mode and sets the GRE key.
<p>Step 6 <code>end</code></p> <p>Example:</p> <pre>Router(config-l3vpn-encap-ip)# end</pre>	Exits L3 VPN encapsulation configuration mode and returns to privileged EXEC mode.
<p>Step 7 <code>show l3vpn encapsulation ip <i>profile-name</i></code></p> <p>Example:</p> <pre>Router# show l3vpn encapsulation ip tunnel encap</pre>	(Optional) Displays the profile health and the underlying tunnel interface.

Configuring BGP and Route Maps

Perform this task to configure BGP and route maps. The following steps also enable you to link the route map to the application template and set up the BGP VPNv4 and VPNv6 exchange so that the updates are filtered through the route map.

SUMMARY STEPS

1. enable
2. configure terminal
3. router bgp *as-number*
4. bgp log-neighbor-changes
5. neighbor *ip-address* remote-as *as-number*
6. neighbor *ip-address* update-source *interface name*
7. address-family ipv4
8. no synchronization
9. redistribute connected
10. neighbor *ip-address* activate
11. no auto-summary
12. exit
13. address-family vpnv4
14. neighbor *ip-address* activate
15. neighbor *ip-address* send-community both
16. neighbor *ip-address* route-map *map-name* in
17. exit
18. address-family vpnv6
19. neighbor *ip-address* activate
20. neighbor *ip-address* send-community both
21. neighbor *ip-address* route-map *map-name* in
22. exit
23. route-map *map-tag* permit *position*
24. set ip next-hop encapsulate l3vpn *profile-name*
25. set ipv6 next-hop encapsulate l3vpn *profile-name*
26. exit
27. exit

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.
	Example:	
	Router> enable	<ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	router bgp <i>as-number</i> Example: Router(config)# router bgp 100	Specifies the number of an autonomous system that identifies the router to other BGP routers and tags the routing information passed along, and enters router configuration mode.
Step 4	bgp log-neighbor-changes Example: Router(config-router)# bgp log-neighbor-changes	Enables logging of BGP neighbor resets.
Step 5	neighbor <i>ip-address</i> remote-as <i>as-number</i> Example: Router(config-router)# neighbor 209.165.200.225 remote-as 100	Adds an entry to the BGP or multiprotocol BGP neighbor table.
Step 6	neighbor <i>ip-address</i> update-source <i>interface name</i> Example: Router(config-router)# neighbor 209.165.200.225 update-source loopback 0	Allows BGP sessions to use any operational interface for TCP connections.
Step 7	address-family ipv4 Example: Router(config-router)# address-family ipv4	Enters address family configuration mode to configure routing sessions that use IPv4 address prefixes.
Step 8	no synchronization Example: Router(config-router-af)# no synchronization	Enables the Cisco IOS software to advertise a network route without waiting for an IGP.

Command or Action	Purpose
<p>Step 9 redistribute connected</p> <p>Example:</p> <pre>Router(config-router-af)# redistribute connected</pre>	<p>Redistributes routes from one routing domain into another routing domain and allows the target protocol to redistribute routes learned by the source protocol and connected prefixes on those interfaces over which the source protocol is running.</p>
<p>Step 10 neighbor ip-address activate</p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 209.165.200.225 activate</pre>	<p>Enables the exchange of information with a BGP neighbor.</p>
<p>Step 11 no auto-summary</p> <p>Example:</p> <pre>Router(config-router-af)# no auto-summary</pre>	<p>Disables automatic summarization and sends subprefix routing information across classful network boundaries.</p>
<p>Step 12 exit</p> <p>Example:</p> <pre>Router(config-router-af)# exit</pre>	<p>Exits address family configuration mode.</p>
<p>Step 13 address-family vpnv4</p> <p>Example:</p> <pre>Router(config-router)# address-family vpnv4</pre>	<p>Enters address family configuration mode to configure routing sessions, such as BGP, that use standard VPNv4 address prefixes.</p>
<p>Step 14 neighbor ip-address activate</p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 209.165.200.225 activate</pre>	<p>Enables the exchange of information with a BGP neighbor.</p>
<p>Step 15 neighbor ip-address send-community both</p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 209.165.200.225 send-community both</pre>	<p>Specifies that a communities attribute, for both standard and extended communities, should be sent to a BGP neighbor.</p>

Command or Action	Purpose
<p>Step 16 <code>neighbor ip-address route-map map-name in</code></p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 209.165.200.225 route-map SELECT_UPDATE_FOR_L3VPN in</pre>	Applies the named route map to the incoming route.
<p>Step 17 <code>exit</code></p> <p>Example:</p> <pre>Router(config-router-af)# exit</pre>	Exits address family configuration mode.
<p>Step 18 <code>address-family vpnv6</code></p> <p>Example:</p> <pre>Router(config-router)# address-family vpnv6</pre>	Enters address family configuration mode to configure routing sessions, such as BGP, that use VPNv6 address prefixes.
<p>Step 19 <code>neighbor ip-address activate</code></p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 209.165.200.252 activate</pre>	Enables the exchange of information with a BGP neighbor.
<p>Step 20 <code>neighbor ip-address send-community both</code></p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 209.165.200.252 send-community both</pre>	Specifies that a communities attribute, for both standard and extended communities, should be sent to a BGP neighbor.
<p>Step 21 <code>neighbor ip-address route-map map-name in</code></p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 209.165.200.252 route-map SELECT_UPDATE_FOR_L3VPN in</pre>	Applies the named route map to the incoming route.
<p>Step 22 <code>exit</code></p> <p>Example:</p> <pre>Router(config-router-af)# exit</pre>	Exits address family configuration mode.

Command or Action	Purpose
<p>Step 23 <code>route-map map-tag permit position</code></p> <p>Example:</p> <pre>Router(config-router)# route-map SELECT_UPDATE_FOR_L3VPN permit 10</pre>	<p>Enters route-map configuration mode and defines the conditions for redistributing routes from one routing protocol into another.</p> <ul style="list-style-type: none"> • The redistribute router configuration command uses the specified map tag to reference this route map. Multiple route maps may share the same map tag name. • If the match criteria are met for this route map, the route is redistributed as controlled by the set actions. • If the match criteria are not met, the next route map with the same map tag is tested. If a route passes none of the match criteria for the set of route maps sharing the same name, it is not redistributed by that set. • The <i>position</i> argument indicates the position a new route map will have in the list of route maps already configured with the same name.
<p>Step 24 <code>set ip next-hop encapsulate l3vpn profile-name</code></p> <p>Example:</p> <pre>Router(config-route-map)# set ip next-hop encapsulate l3vpn my profile</pre>	<p>Indicates that output IPv4 packets that pass a match clause of the route map are sent to the VRF for tunnel encapsulation.</p>
<p>Step 25 <code>set ipv6 next-hop encapsulate l3vpn profile-name</code></p> <p>Example:</p> <pre>Router(config-route-map)# set ip next-hop encapsulate l3vpn tunnel encap</pre>	<p>Indicates that output IPv6 packets that pass a match clause of the route map are sent to the VRF for tunnel encapsulation.</p>
<p>Step 26 <code>exit</code></p> <p>Example:</p> <pre>Router(config-route-map)# exit</pre>	<p>Exits route-map configuration mode and enters global configuration mode.</p>
<p>Step 27 <code>exit</code></p> <p>Example:</p> <pre>Router(config)# exit</pre>	<p>Exits global configuration mode.</p>

Configuration Examples for MPLS VPN over mGRE

- [Example Verifying The MPLS VPN over mGRE Configuration, page 328](#)
- [Example Configuration Sequence For MPLS VPN over mGRE, page 328](#)

Example Verifying The MPLS VPN over mGRE Configuration

Use the following examples to verify that the configuration is working properly:

Cisco Express Forwarding (CEF) Switching

You can verify that CEF switching is working as expected:

```
Router# show ip cef vrf Customer_A tunnel 0

209.165.200.250
/24
    nexthop 209.165.200.251 Tunnel0 label 16
```

Endpoint Creation

You can verify the tunnel endpoint that has been created:

```
Router# show tunnel endpoints tunnel 0

Tunnel0 running in multi-GRE/IP mode
Endpoint transport 209.165.200.251 Refcount 3 Base 0x2AE93F0 Create Time 00:00:42
overlay 209.165.200.254 Refcount 2 Parent 0x2AE93F0 Create Time 00:00:42
```

Adjacency

You can verify that the corresponding adjacency has been created:

```
Router# show adjacency tunnel 0

Protocol Interface          Address
IP         Tunnel0                    209.165.200.251(4)
TAG        Tunnel0                    209.165.200.251(3)
```

Profile Health

You can use **show l3vpn encapsulation profile-name** command to get information on the basic state of the application. The output of this command provides you details on the references to the underlying tunnel.

```
Router# show l3vpn encapsulation ip tunnel encap
Profile: tunnel encap
transport ipv4 source Auto: Loopback0
protocol gre
Tunnel Tunnel0 Created [OK]
Tunnel Linestate [OK]
Tunnel Transport Source (Auto) Loopback0 [OK]
```

Example Configuration Sequence For MPLS VPN over mGRE

This example shows the configuration sequence for MPLS VPN over mGRE:

```
vrf definition Customer A
rd 100:110
```

```

route-target export 100:1000
route-target import 100:1000
!
address-family ipv4
exit-address-family
!
address-family ipv6
exit-address-family
!
!
ip cef
!
ipv6 unicast-routing
ipv6 cef
!
!
l3vpn encapsulation ip sample profile name
transport source loopback 0
protocol gre key 1234
!
!
interface Loopback0
 ip address 209.165.200.252 255.255.255.224
 ip router isis
!
interface Serial2/0
 vrf forwarding Customer A
 ip address 209.165.200.253 255.255.255.224
 ipv6 address 3FFE:1001::/64 eui-64
 no fair-queue
 serial restart-delay 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 209.165.200.254 remote-as 100
 neighbor 209.165.200.254 update-source Loopback0
!
 address-family ipv4
  no synchronization
  redistribute connected
  neighbor 209.165.200.254 activate
  no auto-summary
  exit-address-family
!
 address-family vpnv4
  neighbor 209.165.200.254 activate
  neighbor 209.165.200.254 send-community both
  neighbor 209.165.200.254 route-map SELECT_UPDATE_FOR_L3VPN in
  exit-address-family
!
 address-family vpnv6
  neighbor 209.165.200.254 activate
  neighbor 209.165.200.254 send-community both
  neighbor 209.165.200.254 route-map SELECT_UPDATE_FOR_L3VPN in
  exit-address-family
!
 address-family ipv4 vrf Customer A
  no synchronization
  redistribute connected
  exit-address-family
!
 address-family ipv6 vrf Customer A
  redistribute connected
  no synchronization
  exit-address-family
!
!
route-map SELECT_UPDATE_FOR_L3VPN permit 10
set ip next-hop encapsulate sample profile name
set ipv6 next-hop encapsulate sample profile name

```

Additional References

Related Documents

Related Topic	Document Title
Configuring MPLS Layer 3 VPNs	<i>Cisco IOS Multiprotocol Label Switching Configuration Guide</i>
Dynamic Layer 3 VPNs with multipoint GRE tunnels	<i>Cisco IOS Interface and Hardware Component Configuration Guide</i>
Cisco Express Forwarding	<i>Cisco IOS IP Switching Configuration Guide</i>
Generic routing encapsulation	<i>Cisco IOS Interface and Hardware Component Configuration Guide</i>

Standards

Standard	Title
None	--

MIBs

MIB	MIBs Link
IETF-PPVPN-MPLS-VPN-MIB	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 2547	<i>BGP/MPLS VPNs</i>
RFC 2784	<i>Generic Routing Encapsulation (GRE)</i>
RFC 2890	Key Sequence Number Extensions to GRE
RFC 4023	Encapsulating MPLS in IP or Generic Routing Encapsulation
RFC 4364	<i>BGP/MPLS IP Virtual Private Networks (VPNs)</i>

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/cisco/web/support/index.html</p>

Feature Information for MPLS VPN over mGRE

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 23 Feature Information for MPLS VPN over mGRE

Feature Name	Releases	Feature Information
MPLS VPN over mGRE	12.2(33)SRE 15.1(2)T	<p>This feature provides support to carry MPLS Layer 3 VPN traffic over mGRE. This feature also supports SIP-400 and ES-40 on Cisco 7600 series routers.</p> <p>The following commands were introduced or modified by this feature: <code>l3vpn encapsulation ip protocol gre</code> , <code>show l3vpn encapsulation ip</code> , <code>transport ipv4,set ip next-hop</code> , <code>set ipv6 next-hop</code>.</p>

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks.

Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



IP Tunnel MIBs

This module contains information about MIBs used with interfaces and hardware components. The IP Tunnel MIB feature provides a generic MIB for managing all IPv4- and IPv6-related tunnels, as outlined in RFC 4087, IP Tunnel MIB. Tunneling provides a way to encapsulate arbitrary packets inside a transport protocol. A number of tunneling mechanisms specified by Internet Engineering Task Force (IETF) are implemented by Cisco for both IPv4 and IPv6 environments. Various MIBs are available for managing tunnels.

- [Finding Feature Information, page 333](#)
- [Prerequisites for the IP Tunnel MIB, page 333](#)
- [Restrictions for the IP Tunnel MIB, page 333](#)
- [Information About the IP Tunnel MIB, page 334](#)
- [How to Configure SNMP and Use the IP Tunnel MIB, page 336](#)
- [Additional References, page 337](#)
- [Feature Information for the Tunnel MIB, page 339](#)

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.

Prerequisites for the IP Tunnel MIB

Configure Simple Network Management Protocol (SNMP) on the router on which the IP Tunnel MIB feature is to be used. See the [Configuring the Router to Use SNMP, page 336](#) for more information. For more information on configuring an SNMP server, see the "Configuring SNMP Support" chapter of the Cisco IOS Network Management Configuration Guide.

Restrictions for the IP Tunnel MIB

The IP Tunnel MIB feature supports only tunnels that can be created using the **interface tunnel** command. The IP Tunnel MIB feature does not support Layer 2 Tunnel Protocol (L2TP), Point-to-Point Tunneling Protocol (PPTP), or Multiprotocol Label Switching (MPLS) tunnels.

Information About the IP Tunnel MIB

- [Benefits of the IP Tunnel MIB, page 334](#)
- [MIB Objects Supported by the IP Tunnel MIB, page 334](#)

Benefits of the IP Tunnel MIB

Improved Quality of Networks

Better IP tunnel instrumentation leads to an improvement in the quality of networks and better service delivery. A better quality network allows service providers to deliver a more reliable service.

Increased Reliability

The IP Tunnel MIB allows users of network management systems to set inventory and receive notification about their IP tunnel activity.

The IP Tunnel MIB supports both IPv4 and IPv6 network layers as defined in RFC 3291, and is used to manage IP tunnels implemented in the Cisco IOS software.

The IP Tunnel MIB supports all tunnel types, as well as tunnel creation and destruction capability.

Interoperability with Devices Other Than Cisco Devices

The IP Tunnel MIB works with key network management systems, including those of third-party vendors.

MIB Objects Supported by the IP Tunnel MIB

The following MIB objects are supported by the IP Tunnel MIB feature. For details regarding use of MIB objects, see RFC 4087, IP Tunnel MIB.

Table 24 *Objects Supported by the IP Tunnel MIB*

MIB Object	Description
tunnelIfEntry	Contains information on a particular configured tunnel. You can use the interface tunnel command to set a value for this object.
tunnelIfEncapsMethod	The encapsulation method used by the tunnel. You can use the tunnel mode command to set a value for this object.
tunnelIfHopLimit	Defines the IPv4 time to live (TTL) or IPv6 hop limit to use in the outer IP header. You can use the tunnel ttl command to set a value for this object.

MIB Object	Description
tunnelIfSecurity	Used by the tunnel to secure the outer IP header. The value ipsec indicates that IPsec is used between the tunnel endpoints for authentication or encryption, or both.
tunnelIfTOS	Used by the tunnel to set the high 6 bits (the differentiated services codepoint) of the IPv4 type of service (ToS) or IPv6 traffic class in the outer IP header. You can use the tunnel tos command to set a value for this object.
tunnelIfFlowLabel	Used to set the IPv6 Flow Label value. This object is supported for tunnels over IPv6. The default value for this object is 0.
tunnelIfAddressType	Shows the type of address in the corresponding tunnelIfLocalInetAddress and tunnelIfRemoteInetAddress objects. This object cannot be configured individually through the command-line interface (CLI).
tunnelIfLocalInetAddress	The address of the local endpoint of the tunnel (that is, the source address used in the outer IP header). If the address is unknown, the value is 0.0.0.0 for IPv4 or :: for IPv6. The address type of this object is given by tunnelIfAddressType. You can use the tunnel source command to set a value for this object.
tunnelIfRemoteInetAddress	The address of the remote endpoint of the tunnel (that is, the destination address used in the outer IP header). If the address is unknown or the tunnel is not a point-to-point link (for example, a 6-to-4 tunnel), the value is 0.0.0.0 for tunnels over IPv4 or :: for tunnels over IPv6. The address type of this object is given by tunnelIfAddressType. You can use the tunnel destination command to set a value for this object.
tunnelIfEncapsLimit	Shows the maximum number of additional encapsulations permitted for packets undergoing encapsulation at this node. A value of -1 indicates that no limit is present (except as result of packet size).
tunnelInetConfigEntry	Contains information on a particular configured tunnel. There will be only one entry for multipoint tunnels and for tunnels that have the remote inet address 0.0.0.0 for IPv4 or :: for IPv6. Only generic routing encapsulation (GRE)/IP and GRE/IPv6 tunnels are created through the MIB.
tunnelInetConfigIfIndex	Shows the value of ifIndex corresponding to the tunnel interface. A value of 0 is not legal in the active state and means that the interface index has not yet been assigned.
tunnelInetConfigStatus	Used to create or delete table entries in the MIB table. You can use the interface tunnel to set a value for this object.

MIB Object	Description
tunnelInetConfigStorageType	Indicates the storage type. Only a nonvolatile storage value is supported.

How to Configure SNMP and Use the IP Tunnel MIB

- [Configuring the Router to Use SNMP, page 336](#)

Configuring the Router to Use SNMP



Note

Some of the tasks in this section include examples of the SNMP CLI syntax used to set configuration parameters on the router and to read values from MIB objects on the router. These SNMP CLI syntax examples are taken from a Linux workstation using public domain SNMP tools. The SNMP CLI syntax for your workstation might be different. See the documentation that was provided with your SNMP tools for the correct syntax for your network management workstation.

Before you can use the IP Tunnel MIB feature, you must first configure the router to support SNMP. Perform this task to enable SNMP on the router.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server community *string1* ro**
4. **snmp-server community *string2* rw**
5. **end**

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 configure terminal Example: Router# configure terminal	Enters global configuration mode.

Command or Action	Purpose
<p>Step 3 <code>snmp-server community <i>string1</i> ro</code></p> <p>Example:</p> <pre>Router(config)# snmp-server community public ro</pre>	<p>Sets up the community access string to permit access to SNMP.</p> <ul style="list-style-type: none"> The <i>string1</i> argument is a community string that consists of from 1 to 32 alphanumeric characters and functions much like a password, permitting access to the SNMP protocol. Blank spaces are not permitted in the community string. The ro keyword specifies read-only access. SNMP management stations using this string can retrieve MIB objects. <p>Note The SNMP community read-only (RO) string for the examples is public. You should use a more complex string for this value in your configuration.</p>
<p>Step 4 <code>snmp-server community <i>string2</i> rw</code></p> <p>Example:</p> <pre>Router(config)# snmp-server community private rw</pre>	<p>Sets up the community access string to permit access to SNMP.</p> <ul style="list-style-type: none"> The <i>string2</i> argument is a community string that consists of from 1 to 32 alphanumeric characters and functions much like a password, permitting access to the SNMP protocol. Blank spaces are not permitted in the community string. The rw keyword specifies read-write access. SNMP management stations using this string can retrieve and modify MIB objects. <p>Note The SNMP community read-write (RW) string for the examples is private. You should use a more complex string for this value in your configuration.</p>
<p>Step 5 <code>end</code></p> <p>Example:</p> <pre>Router(config)# end</pre>	<p>Exits the current configuration mode and returns to privileged EXEC mode.</p>

- [What to Do Next, page 337](#)

What to Do Next

To implement the IP Tunnel MIB, you must configure a tunnel. For information on configuring tunnels, see the "Implementing Tunnels" chapter in the Cisco IOS Interface and Hardware Component Configuration Guide.

To debug or troubleshoot any issues related to configuring the IP Tunnel MIB through SNMP, use the `debug snmp tunnel-mib` command. For information on this command see Cisco IOS Interface and Hardware Component Command Reference.

Additional References

Related Documents

Related Topic	Document Title
SNMP commands, complete command syntax, command reference, command history, defaults, defaults, usage guidelines, and examples	<i>Cisco IOS Network Management Command Reference</i>
Configuring SNMP Support	<i>Cisco IOS Network Management Configuration Guide</i>
Implementing tunnels	<i>Cisco IOS Interface and Hardware Component Configuration Guide</i>

Standards

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

MIBs

MIB	MIBs Link
IP Tunnel MIB	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

RFC	Title
RFC 4087	IP Tunnel MIB

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for the Tunnel MIB

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 25 Feature Information for the IP Tunnel MIB

Feature Name	Releases	Feature Information
IP Tunnel MIB	12.2(33)SRB 12.2(1st)SY 12.2(44)SG 12.2(33)SRD 15.0(1)M Cisco IOS XE 3.1.0SG	The IP Tunnel MIB provides a generic MIB for managing all IPv4- and IPv6-related tunnels, as outlined in RFC 4087 IP Tunnel MIB.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



Managing Dial Shelves

This chapter discusses configuration and monitoring tasks for dial shelves and dial shelf controllers, particularly on Cisco AS5800 Universal Access Servers.

To identify the hardware platform or software image information associated with a feature, use Cisco Feature Navigator on Cisco.com to search for information about the feature.

For additional information about the technologies in this chapter, see the following publications:

- *Dial and System Management Commands for the Cisco AS5800* (This document is available online only.)
- *Cisco AS5800 Access Server Software ICG*
- *Cisco IOS Dial Technologies Configuration Guide* and *Cisco IOS Dial Technologies Command Reference* (Release 12.2)

For hardware technical descriptions and information about installing interfaces, refer to the hardware installation and configuration publication for your product. For a complete description of dial shelf management commands in this chapter, refer to the *Cisco IOS Interface and Hardware Component Command Reference*. To locate documentation of other commands that appear in this chapter, use the master commands list or search online.

- [Finding Feature Information, page 341](#)
- [Dial Shelf Management Task List, page 341](#)
- [Understanding Shelf Architecture and DSIP, page 342](#)
- [Maintaining Shelves, page 342](#)
- [Maintaining Dial Shelf Controllers, page 345](#)
- [Troubleshooting Using DSIP, page 347](#)

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.

Dial Shelf Management Task List

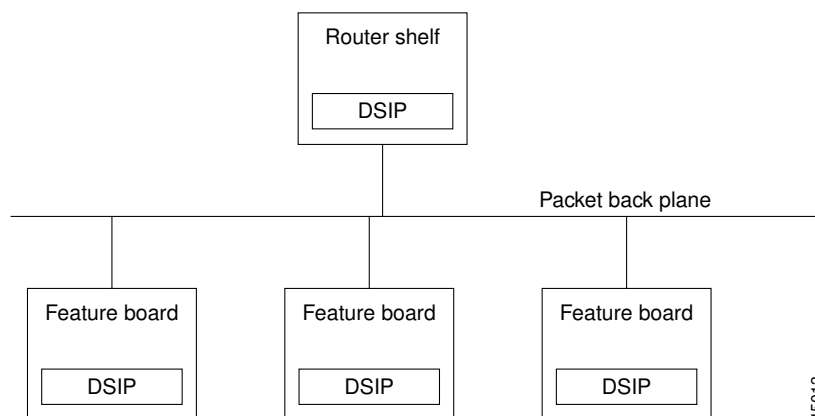
To manage dial shelves, perform the tasks in the following sections:

Understanding Shelf Architecture and DSIP

The Cisco AS5800 is a rack-mounted system consisting of a router shelf and a dial shelf. The dial shelf contains trunk cards, modem cards, and dial shelf controller (DSC) cards. The trunk cards and modem cards are referred to collectively as feature boards. Slots 0 through 11 of the dial shelf are reserved for feature boards, while slots 12 and 13 are reserved for the DSC cards. The AS5800 series supports the use of a single router shelf or two router shelves (split-shelf configuration), and the use of a single DSC or two DSCs (DSC redundancy) for backup purposes.

Dial Shelf Interconnect Protocol (DSIP) is used for communication between router shelf and dial shelf on an AS5800. The figure below diagrams the components of the architecture. DSIP communicates over the packet backplane via the dial shelf interconnect (DSI) cable.

Figure 32 DSIP Architecture in the Cisco AS5800



Maintaining Shelves

Perform the tasks described in the following sections to perform the respective configuration tasks:

- [Configuring the Shelf IDs, page 342](#)
- [Configuring the Shelf IDs, page 344](#)
- [Executing Commands Remotely, page 345](#)

Configuring the Shelf IDs

The Cisco AS5800 consists of one or more router shelves and a dial shelf. Shelf ID numbers and port numbers are used to identify specific components in your system. The default shelf number is 0 for the router shelf and 1 for the dial shelf.

Normally you do not need to change the shelf IDs; however, if you do, we recommend that you change the shelf number when you initially access the setup facility. For information on the setup facility, refer to the *Cisco AS5800 Universal Access Server Software Installation and Configuration Guide*.

**Caution**

You must reload the Cisco AS5800 for the new shelf number to take effect. Because the shelf number is part of the interface names when you reload, all NVRAM interface configuration information is lost.

If you are booting the router shelf from the network (netbooting), you can change the shelf numbers using the **shelf-id** command. Perform the following steps beginning in EXEC mode.

SUMMARY STEPS

1. **copy startup-configure tftp**
2. **configure terminal**
3. **shelf-id *number* router-shelf**
4. **shelf-id *number* dial-shelf**
5. **exit**
6. **copy running-config startup-config**
7. **show version**
8. **reload**
9. Type "yes" to the "save config" prompt.
10. Configure one interface so that router shelf has connectivity to the server with the configuration.
11. **copy tftp startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	copy startup-configure tftp	Saves your current configuration. Changing the shelf number removes all interface configuration information when you reload the Cisco AS5800.
Step 2	configure terminal	Enters configuration mode.
Step 3	shelf-id <i>number</i> router-shelf	Specifies the router shelf ID.
Step 4	shelf-id <i>number</i> dial-shelf	Specifies the dial shelf ID.
Step 5	exit	Exits configuration mode.
Step 6	copy running-config startup-config	(Optional) Saves your configuration.
Step 7	show version	Verifies that the correct shelf number will be changed after the next reload.
Step 8	reload	Reloads the Cisco AS5800.
Step 9	Type "yes" to the "save config" prompt.	--
Step 10	Configure one interface so that router shelf has connectivity to the server with the configuration.	--
Step 11	copy tftp startup-config	Because changing the shelf number removes all interface configuration information when you reload the Cisco AS5800, edit the configuration file saved in Step 1 and download it.

Configuring the Shelf IDs

If you are booting the router shelf from flash memory, perform the following steps beginning in EXEC mode.

SUMMARY STEPS

1. Do one of the following:
 - **copy running-config tftp**
 -
 -
 - **copy startup-config tftp**
2. **configure terminal**
3. **shelf-id *number* router-shelf**
4. **shelf-id *number* dial-shelf**
5. **exit**
6. **copy running-config startup-config**
7. **show version**
8. Edit the configuration file saved in Step 1.
9. **copy tftp startup-config**
10. **reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Do one of the following: <ul style="list-style-type: none"> • copy running-config tftp • • • copy startup-config tftp 	Saves your current (latest) configuration to a server.
Step 2	configure terminal	Enters configuration mode.
Step 3	shelf-id <i>number</i> router-shelf	Configures the router shelf ID.
Step 4	shelf-id <i>number</i> dial-shelf	Configures the dial shelf ID.
Step 5	exit	Exits configuration mode.
Step 6	copy running-config startup-config	(Optional) Saves your configuration. If this step is skipped, type "No" to the 'save configuration' prompt.
Step 7	show version	Allows verification that the correct shelf number will be changed after the next reload.
Step 8	Edit the configuration file saved in Step 1.	--
Step 9	copy tftp startup-config	Copies the edited configuration to NVRAM on the Cisco AS5800.

Command or Action	Purpose
Step 10 reload	Reloads the system.

Executing Commands Remotely

It is possible to connect directly to the system console interface on the DSC to execute dial shelf configuration commands, but this is not recommended. All commands necessary for dial shelf configuration, including show and debug tasks, can be executed remotely through the router console. A special command called **execute-on** is provided for this purpose. This command enables a special set of Exec mode commands to be executed on the router or the dial shelf. This command is a convenience that avoids connecting the console to the DSC. For a list of commands that you can execute using **execute-on**, see the complete command description in the *Cisco IOS Configuration Fundamentals Command Reference*.

To enter a command that you wish to execute on a specific card installed in the dial shelf while logged onto the router shelf console, use the following privileged EXEC mode commands.

Command	Purpose
execute-on slot <i>slot command</i>	Executes a command from the router shelf on a specific card in the dial shelf.
execute-on all <i>command</i>	Executes a command from the router shelf on all cards in the dial shelf.

Maintaining Dial Shelf Controllers

The DSC card provides the following:

- Master clock for the dial shelf
- Fast Ethernet link to the router shelf
- Environmental monitoring of the feature boards
- Bootstrap images on start-up for the feature boards

The Cisco AS5800 dial shelf can contain two DSC cards. With two DSC cards present, DSC redundancy automatically provides for one DSC to act as a backup to the active one. This redundancy feature is implemented to increase system availability by preventing loss of service in the event of the failure of one of the DSCs. The redundancy is intended to be transparent to most Cisco AS5800 software (redundancy is supported at or below the DSIP layer). Software modules using the DSIP services are generally not aware of nor need to take part in the management of dual DSCs.

- [Configuring Clocks, page 346](#)

- [Monitoring and Maintaining DSCs and Feature Boards, page 346](#)

Configuring Clocks

The TDM bus in the backplane on the dial shelf must be synchronized to the T1/E1 clocks on the trunk cards. The Dial Shelf Controller (DSC) card on the dial shelf provides hardware logic to accept multiple clock sources as input and use one of them as the primary source to generate a stable, PPL synchronized output clock. The input clock can be any of the following sources:

- Trunk port in slots 0 through 5--up to 12 can be selected (2 per slot)
- An external T1 or E1 clock source fed directly through a connector on the DSC card
- A free-running clock from an oscillator in the clocking hardware on the DSC card

For dual (redundant) DSC cards, the external DSC clocking port should be configured so that the clock signal fed into both DSCs is identical.

To configure the clock source and priority of the clock source used by the TDM bus, perform one or more of the following steps, beginning in global configuration mode.

SUMMARY STEPS

1. `dial-tdm-clock priority number trunk-slot slot port number`
2. `dial-tdm-clock priority number freerun`
3. `dial-tdm-clock priority number external {e1 | t1} [120ohm]`
4. `exit`
5. `copy running-config startup-config`
6. `show dial-shelf clocks`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>dial-tdm-clock priority number trunk-slot slot port number</code>	Configure the priority of the trunk card clock.
Step 2	<code>dial-tdm-clock priority number freerun</code>	Configure the priority of the free running clock.
Step 3	<code>dial-tdm-clock priority number external {e1 t1} [120ohm]</code>	Configure the priority of the T1 or E1 external clock.
Step 4	<code>exit</code>	Exit configuration mode.
Step 5	<code>copy running-config startup-config</code>	Save your configuration.
Step 6	<code>show dial-shelf clocks</code>	Verify the clocking priorities.

Monitoring and Maintaining DSCs and Feature Boards

Use the following commands in privileged EXEC mode to swap dial shelf cards or to troubleshoot the dial shelf cards from the router shelf.

Command	Purpose
<pre>hw-module slot shelf-id / slot-number {start stop}</pre>	Stops a DSC remotely from the router console or restarts the DSC if it has been stopped.
<pre>hw-module slot shelf-id / slot-number reload</pre>	Reloads the specified feature board. This command can be used instead of a manual online insertion and removal (OIR) to reload and power-cycle a feature board. Note that this command cannot be applied to DSCs.
<pre>show redundancy [history]</pre>	Displays the current or history status for redundant DSC.
<pre>ui clk hub} debug redundancy {all </pre>	Use this debug command if you need to collect events for troubleshooting, selecting the appropriate required key word.
<pre>show debugging</pre>	Lists the debug commands that are turned on, including that for redundant DSC.

Troubleshooting Using DSIP

There are a number of **show** commands available to aid in troubleshooting dial shelves. Use any of the following EXEC mode commands to monitor DSI and DSIP activity.

Command	Purpose
<pre>clear dsip tracing</pre>	Used to clear tracing statistics for the Distributed System Interconnect Protocol (DSIP).
<pre>show dsip</pre>	Displays all information about the Distributed System Interconnect Protocol (DSIP).
<pre>show dsip clients</pre>	Displays information about Distributed System Interconnect Protocol (DSIP) clients.

Command	Purpose
<code>show dsip nodes</code>	Displays information about the processors running the Distributed System Interconnect Protocol (DSIP).
<code>show dsip ports</code>	Displays information about local and remote ports.
<code>show dsip queue</code>	Displays the number of messages in the retransmit queue waiting for acknowledgment.
<code>show dsip tracing</code>	Displays Distributed System Interconnect Protocol (DSIP) tracing buffer information.
<code>show dsip transport</code>	Displays information about the Distributed System Interconnect Protocol (DSIP) transport statistics for the control/data and IPC packets and registered addresses.
<code>show dsip version</code>	Displays Distributed System Interconnect Protocol (DSIP) version information.

The privileged EXEC mode `show dsi` command can also be used to troubleshoot, as it displays the status of the DSI adapter, which is used to physically connect the router shelf and the dial shelf to enable DSIP communications.

The following is an example troubleshooting scenario:

Problem: The router shelf boots, but there is no communication between the router and dial shelves.

SUMMARY STEPS

1. Run the `show dsip transport` command.
2. Check the "DSIP registered addresses" column. If there are zero entries here, there is some problem with the Dial Shelf Interconnect (DSI). Check if the DSI is installed in the router shelf.
3. If there is only one entry and it is our own local address, then first sanity check the physical layer. Make sure that there is a physical connection between the RS and DS. If everything is fine from a cabling point of view, go to [Troubleshooting Using DSIP, page 347](#).
4. Check the DSI health by issuing the `show dsi` command. This gives a consolidated output of DSI controller and interface. Check for any errors like runts, giants, throttles and other usual FE interface errors.

DETAILED STEPS

- Step 1** Run the **show dsip transport** command.
- Step 2** Check the "DSIP registered addresses" column. If there are zero entries here, there is some problem with the Dial Shelf Interconnect (DSI). Check if the DSI is installed in the router shelf.
- Step 3** If there is only one entry and it is our own local address, then first sanity check the physical layer. Make sure that there is a physical connection between the RS and DS. If everything is fine from a cabling point of view, go to [Troubleshooting Using DSIP, page 347](#).
- Step 4** Check the DSI health by issuing the **show dsi** command. This gives a consolidated output of DSI controller and interface. Check for any errors like runts, giants, throttles and other usual FE interface errors.
-

Diagnosis: If an entry for a particular dial shelf slot is not found among the registered addresses, but most of other card entries are present, the problem is most likely with that dial shelf slot. The DSI hardware on that feature board is probably bad.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



Router-Shelf Redundancy for the Cisco AS5800

This document describes router-shelf redundancy for the Cisco AS5800 universal access server. It includes the following sections:

- [Finding Feature Information, page 351](#)
- [Feature Overview, page 351](#)
- [Prerequisites, page 354](#)
- [Configuration Examples, page 354](#)
- [Additional References, page 357](#)
- [Feature Information for Router-Shelf Redundancy for the Cisco AS5800, page 358](#)
- [Glossary, page 358](#)

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.

Feature Overview

This feature provides router-shelf redundancy by using a second router shelf that automatically takes over the other shelf's dial-shelf cards (DSCs) if it appears that the other router shelf has died. Failover is disruptive in that there is no attempt to maintain calls that were established on the failing router shelf; the DSCs controlled by the failing router shelf are restarted under control of the backup router shelf and hence become available again.

Two router shelves are connected to the same DSC (as in split mode), but with only one router shelf active at a time. Both router shelves are configured for normal mode as opposed to split mode. Each router shelf contains the same configuration, being whatever configuration is appropriate for the full set of DSCs. The active router shelf controls all the DSCs, while the other router shelf functions purely as a backup. If the active router shelf fails, all DSCs restart under the control of the backup router shelf, which then functions as the active router shelf.

Only one router shelf has control of the DSCs at a time; the other keeps trying to take control but is unable to, and does not interfere with operation of the active router shelf. If, however, the active router shelf crashes, then it relinquishes control of all DSCs to the other router shelf, which restarts the DSCs and

commences normal operation. If the crashed router shelf recovers or is restarted, it does not take back control of the DSCs, but instead functions as a backup, and takes control again only if the other router shelf fails.

External interfaces cannot share the same IP address between the two routers shelves, to prevent duplicate IP address errors.


Note

Triggers for a failover to occur are those that lead to a hub switchover. The main trigger is loss of the link between the active router shelf (the one with control of the cards) and its DSC *as detected by link monitoring on the DSC*. Any router-shelf failures that do not result in this link going down do not cause failover—for example, the active router’s egress interface going down does not trigger failover. Conversely, any temporary loss of the link between the active router and a DSC *does* cause failover, even if the router shelf itself does not crash and connectivity is quickly reestablished—for example, if the BIC cable is knocked out and then quickly replaced. In addition, failover is triggered if a DSC connected to the active router shelf goes down and fails to recover within 90 seconds.

- [Additional Considerations, page 352](#)
- [Benefits, page 353](#)
- [Restrictions, page 353](#)
- [Related Features and Technologies, page 354](#)

Additional Considerations

- [System Controller, page 352](#)
- [Load Sharing, page 352](#)
- [Hitless Redundancy, page 353](#)
- [Network Management, page 353](#)

System Controller

When a system controller is used with a redundant router shelf, router-shelf failover should look like a single router shelf going down briefly and then recovering. With the current system-controller code, this does not work. The Cisco SC3640 expects only a single router shelf to be configured with any given shelf ID. To get around this, a router in backup mode must be prevented from sending Session Definition Protocol (SDP) packets to the Cisco SC3640. In addition, the SDP packets sent by the router shelves to the Cisco SC3640 currently include a field identifying the MAC address of the sending router. The Cisco SC3640 stores this MAC address and, if it subsequently receives another SDP packet containing the same shelf ID but not the same MAC address, it concludes that multiple routers are configured with the same shelf ID and treats this as an error. This is precisely the situation after a failover, when the backup router shelf starts sending SDP packets with the same shelf ID but different MAC address.

To get around this, you must configure a failover group code—an integer that identifies a redundant pair of router shelves. Each member of the pair must be configured with the same group code. When failover mode is enabled, this group code is sent in place of the router MAC address. These changes are all made to the system-controller code that runs on the router shelf itself, rather than on the Cisco SC3640.

Load Sharing

There is no load sharing between the two routers shelves—no calls can go through the backup router shelf. One disadvantage of this is that you cannot split the load between the routers to reduce the number of calls

that are lost when a router crashes. There are, however, also some advantages: with load sharing, you must ensure that each router can support the entire dial shelf, since upon failure of a router shelf the surviving router shelf owns all the dial-shelf resources, and therefore has a sudden change in the amount of traffic it is supporting. If care had not been taken to test under failover conditions, at full load the surviving router shelf might be overwhelmed, and perhaps provide degraded service. With a redundant router shelf instead acting purely as a standby, provided that the backup router shelf is the same model as the active router shelf, the load is unchanged after switchover--so no change is expected to the router-shelf performance.

Having a single router shelf active at a time is simpler, and makes it easier to support failover when dealing with external servers such as signaling controllers for SS7, RPMS server, and system controllers.

Hitless Redundancy

When router-shelf failover occurs, all calls associated with the failed router shelf are lost. To maintain calls through router-shelf failure requires mirroring call state and fast failure detection. This shelf-redundancy feature ensures only that resources (particularly trunk lines) do not remain unusable while the router shelf that was controlling them is down.

Network Management

Minimal Simple Network Management Protocol (SNMP) support is provided--a trap is issued when failover occurs, and SNMP variables indicate whether a router shelf is active or on standby. An existing MIB--CISCO-C8500-REDUNDANCY-MIB--defines a suitable trap for issuing on failover.

Benefits

When an active router shelf in a Cisco AS5800 loses communication with its DSC, a backup router shelf can be invoked to automatically take over DSCs controlled by the lost router shelf. This backup method, called redundancy, is provided on the Cisco AS5800 to prevent a single point of failure, subsequent downtime, and user intervention to resolve unrecoverable hardware faults.

Restrictions

Router Shelves

Two router shelves of the same model and configuration must be available for this feature to operate.

External Servers

Although some of the failover functionality exists in the existing code base, ensure that the various external servers that can run with the Cisco AS5800 still function when redundant router shelves are used. The servers of concern at the moment are RPMS, SS7, and System Controller; the first two are discussed below:

- Resource pool management server (RPMS). For Resource Pool Management (RPM) to work, the resource pool manager server (RPMS) must be configured with the same information for both router shelves.
- Signaling System 7 (SS7). In an SS7 setup, the call signaling comes through an external Cisco SC2200 unit rather than directly from the switch over the trunk line (as for CAS and ISDN). For call signaling to work after failover, both router shelves must be connected to the Cisco SC2200 using the SS7 Redundant Link Manager (RLM) feature, which was intended to provide redundant links between a single router shelf and the signaling controller. RLM links must be configured from both the active

and standby router shelves--the change of router shelves will look like a change from one redundant link to another.

Related Features and Technologies

RSC Handover Redundancy

The Router Shelf Controller Handover Redundancy feature that is available on the Cisco AS5850 is similar to Router Shelf Redundancy on the Cisco AS5800.

Prerequisites

- Ensure that your network is up and running.
- Test each router shelf to verify connectivity.

Configuration Examples

- [Configuring the Cisco AS5800 for Shelf Redundancy, page 354](#)
- [Configuring the Shelf Redundancy Feature, page 355](#)
- [Verifying Shelf Redundancy, page 355](#)

Configuring the Cisco AS5800 for Shelf Redundancy

SUMMARY STEPS

1. Router(config)# **redundancy**
2. Router(config-red)# **failover group-number** *group-code*

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config)# redundancy	Enters configuration-redundancy mode.
Step 2	Router(config-red)# failover group-number <i>group-code</i>	Configures router-shelf failover. Note Use the same <i>group-code</i> argument for both routers. This code is used when the system controller is used and it identifies the two routers as effectively representing the same set of dial-shelf resources.

Connect to each router shelf in turn and enter these commands. Treat router shelves as if they are connected as a split dial-shelf configuration.

**Note**

This configuration by itself is not enough for successful failover to occur. Because there is no automatic synchronization of configuration between router shelves, you must configure each router shelf separately. Typically the two router shelves, active and backup, must have the same configuration except for the IP address on egress interfaces.

**Note**

Since configuration is error prone, test the backup router shelf's configuration to ensure that errors are not discovered only when the active router shelf fails in a production environment.

Configuring the Shelf Redundancy Feature

SUMMARY STEPS

1. Router(config)# **redundancy**
2. Router(config-red)# **failover group-number group-number**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config)# redundancy	Enters configuration-redundancy mode.
Step 2	Router(config-red)# failover group-number group-number	Sets the router shelf to failover mode. You must enter this command on both router shelves.

Verifying Shelf Redundancy

Use the **show redundancy** command to verify whether the "Shelf is redundant" string is displayed on a redundancy-enabled access server, as illustrated below:

```
Router# show redundancy
T1 1/0 is up.
  Applique type is Channelized T1
  Cablelength is long gain36 0db
  No alarms detected.
  alarm-trigger is not set
  Version info of slot 3: HW: 256, PLD Rev: 1
  Framer Version: 0x8
  Manufacture Cookie Info:
  EEPROM Type 0x0001, EEPROM Version 0x01, Board ID 0x02,
  Board Hardware Version 1.0, Item Number 32-0-00,
  Board Revision 00, Serial Number 12059498,
  PLD/ISP Version <unset>, Manufacture Date 19-Jun-1999.
  Framing is ESF, Line Code is AMI, Clock Source is Line.
  Trunk setting is rbs-zero.
  Data in current interval (619 seconds elapsed):
    380 Line Code Violations, 171 Path Code Violations
    0 Slip Secs, 0 Fr Loss Secs, 137 Line Err Secs, 10 Degraded Mins
    137 Errored Secs, 21 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
  Total Data (last 24 hours)
    203903 Line Code Violations, 27284 Path Code Violations,
    7 Slip Secs, 531 Fr Loss Secs, 18414 Line Err Secs, 1431 Degraded Mins
```

The following example shows output from two router shelves configured as a failover pair. The active router shelf is initially RouterA. The commands **show redundancy history** and **show redundancy** have

been issued. The **show redundancy** command shows that failover is enabled and shows the configured group number. The **show redundancy** command also shows that this router shelf is the active one of the pair. Compare this output with that from the backup router shelf (RouterB) further below.

**Note**

When RouterA is reloaded, thereby forcing a failover, new entries are shown on RouterB when a **show redundancy history** command is issued after failover has occurred.

Log from the First Router Shelf (RouterA):

```
RouterA#
RouterA# show redundancy history
DSC Redundancy Status Change History:
010215 18:17 Slot -1 DSC:Failover configured -> ACTIVE role by default.
010215 18:18 Slot -1 DSC:Failover -> BACKUP role.
010215 18:18 Slot 12 DSC:Failover -> ACTIVE role.
010215 18:18 Slot 12 DSC:Hub, becoming active - arb timeout
RouterA#
RouterA# show redundancy
failover mode enabled, failover group = 32
Currently ACTIVE role.
DSC in slot 12:
Hub is in 'active' state.
Clock is in 'active' state.
No connection to slot 13
RouterA#
RouterA# reload
Proceed with reload? [confirm]
*Feb 15 20:19:11.059:%SYS-5-RELOAD:Reload requested
System Bootstrap, Version 11.1(13)CA, EARLY DEPLOYMENT RELEASE SOFTWARE (fc1)
Copyright (c) 1997 by cisco Systems, Inc.
C7200 processor with 131072 Kbytes of main memory
rommon 1 >
```

Log from the Second Router Shelf (RouterB):

```
RouterB#
RouterB# show redundancy
failover mode enabled, failover group = 32
Currently BACKUP role.
No connection to slot 12
DSC in slot 13:
Hub is in 'backup' state.
Clock is in 'backup' state.
RouterB#
*Feb 16 03:24:53.931:%DSC_REDUNDANCY-3-BICLINK:Switching to DSC 13
*Feb 16 03:24:53.931:%DSC_REDUNDANCY-3-BICLINK:Failover:changing to active mode
*Feb 16 03:24:54.931:%DIAL13-3-MSG:
02:32:06:%DSC_REDUNDANCY-3-EVENT:Redundancy event:LINK_FAIL from other DSC
*Feb 16 03:24:55.491:%OIR-6-INSCARD:Card inserted in slot 12, interfaces administratively
shut down
*Feb 16 03:24:58.455:%DIAL13-3-MSG:
02:32:09:%DSC_REDUNDANCY-3-EVENT:Redundancy event:LINK_FAIL from other DSC
*Feb 16 03:25:04.939:%DIAL13-0-MSG:
RouterB# show redundancy
failover mode enabled, failover group = 32
Currently ACTIVE role.
No connection to slot 12
DSC in slot 13:
Hub is in 'active' state.
Clock is in 'backup' state.
RouterB# show redundancy history
DSC Redundancy Status Change History:
010216 03:09 Slot -1 DSC:Failover configured -> BACKUP role.
010216 03:24 Slot 13 DSC:Failover -> ACTIVE role.
010216 03:24 Slot 13 DSC:Hub, becoming active - D12 linkfail
```

```

010216 03:24 Slot 13 DSC:Hub, becoming active - D12 linkfail
RouterB#
*Feb 16 03:26:14.079:%DSIPPF-5-DS_HELLO:DSIP Hello from shelf 47 slot 1 Succeeded
*Feb 16 03:26:14.255:%DSIPPF-5-DS_HELLO:DSIP Hello from shelf 47 slot 3 Succeeded
*Feb 16 03:26:14.979:%DSIPPF-5-DS_HELLO:DSIP Hello from shelf 47 slot 10 Succeeded

```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
<i>AS5800 Operations, Administration, Maintenance, and Provisioning (OAM&P) Guide</i>	http://www.cisco.com/univercd/cc/td/doc/product/access/acs_serv/as5800/sw_conf/58_oamp/index.htm
Cisco IOS Dial Technologies Configuration Guide	http://www.cisco.com/univercd/cc/cc/td/doc/product/software/ios122/122cgcr/fdial_c/index.htm
Cisco IOS Dial Technologies Command Reference	http://www.cisco.com/univercd/cc/cc/td/doc/product/software/ios122/122cgcr/fdial_r/index.htm
Cisco SS7 Interconnect for Access Servers Solution	http://www.cisco.com/univercd/cc/td/doc/product/access/sc/rel7/soln/das22/index.htm
Cisco SS7 Dial Access Solution System Integration	http://www.cisco.com/univercd/cc/td/doc/product/access/sc/r1/
Cisco SC2200 Signaling Controller documentation	http://www.cisco.com/univercd/cc/td/doc/product/access/sc/r2/

Standards

Standard	Title
--	

MIBs

MIB	MIBs Link
<ul style="list-style-type: none"> CISCO-C8500-REDUNDANCY-MIB 	<p>To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL:</p> <p>http://www.cisco.com/go/mibs</p>

RFCs

RFC	Title
--	

Feature Information for Router-Shelf Redundancy for the Cisco AS5800

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 26 *Feature Information for AToM NSF Any Transport over MPLS and AToM Graceful Restart*

Feature Name	Releases	Feature Information
Router-Shelf Redundancy for the Cisco AS5800	12.1(5) XV1	This feature was introduced.
Router-Shelf Redundancy for the Cisco AS5800	12.2(11)T	This feature was integrated into Cisco IOS Release 12.2(11)T and support was added for the the Cisco AS5800 platform.

Glossary

DSC --dial-shelf controller.

DSIP --Dial Shelf Interconnection Protocol.

RLM --redundant link manager.

RPM --resource pool management.

RPMS --resource pool manager server.

SDP --Session Definition Protocol.

SNMP --Simple Network Management Protocol.

SS7 --Signaling System 7.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks.

Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



Route-Switch-Controller Handover Redundancy on the Cisco AS5850

Feature History

Release	Modification
12.2(2)XB1	This feature was introduced on the Cisco AS5850.
12.2(11)T	This feature was integrated into Cisco IOS Release 12.2(11)T and support was added for the Cisco AS5850 platform.

This document describes the Route-Switch-Controller Handover Redundancy feature on the Cisco AS5850. It includes the following sections:

- [Finding Feature Information, page 361](#)
- [Feature Overview, page 362](#)
- [Supported Platforms, page 364](#)
- [Supported Standards MIBs and RFCs, page 364](#)
- [Prerequisites, page 365](#)
- [Configuration Tasks, page 365](#)
- [Configuration Examples, page 373](#)
- [Command Reference, page 379](#)
- [Glossary, page 379](#)

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.

Feature Overview

Route-Switch-Controller Handover Redundancy on the Cisco AS5850, with its provision of handover-split mode, provides the first phase of high availability to the Cisco AS5850 platform.

If your gateway contains two route-switch-controller (RSC) cards, you can configure your Cisco AS5850 into either of two split modes: classic split or handover split.

Classic-Split Mode

Classic-split (the default) mode maximizes system throughput by splitting slots between two RSCs. Each RSC controls a certain set of slots (slots 0-5 are owned by the RSC in slot 6 and slots 8-13 are owned by the RSC in slot 7), and operates as though slots other than those that it controls contain no cards because those cards are controlled by the other RSC. Configuration on each RSC affects only the slots owned by that RSC. Calls on a failed RSC are lost, but calls on the functioning RSC continue normally. Operating a Cisco AS5850 in classic-split mode is the same as having two Cisco AS5850s, each with a separate set of cards.

Handover-Split Mode

Handover-split mode maximizes system availability by allowing an RSC to automatically take control of the slots, cards, and calls of the other RSC should that other RSC fail. Each RSC is configured identically as appropriate for the full set of cards. During normal operation, both RSCs are active, handling their own slots, cards, and calls just as in classic-split mode. Should an RSC fail, the other RSC takes over control of the failed RSC's slots, goes into extraload state, restarts the failed RSC's cards, and handles newly arrived calls on those cards--although calls on the failed RSC are lost at the moment of failure. The failed RSC, should it recover or be restarted, remains in standby state until you instruct the active RSC to hand back its newly acquired slots to the standby RSC. This is, in effect, split dial shelf with handover capability.

Alternately, to use system resources most efficiently, you can operate with one of the two RSCs initially and intentionally in extraload state. In this configuration, RSCA initially controls all slots in the chassis and RSCB is in standby mode, ready to take over should RSCA fail. This allows you to overcome the limits of normal classic-split mode in which, because only six slots are available per RSC, an optimal combination of trunk and DSP cards is difficult to achieve. For more information on performance loads, see the [Restrictions, page 363](#) section.

- [Benefits, page 362](#)
- [Restrictions, page 363](#)
- [Related Features and Technologies, page 363](#)
- [Related Documents, page 363](#)

Benefits

High Availability

RSC Handover Redundancy for the Cisco AS5850, enabled in handover-split mode, eliminates any single point of failure, subsequent downtime, and required user intervention to resolve unrecoverable hardware faults. This improves service availability and reduces both service-affecting time and service interruption.

Restrictions

RSC Card Requirements

You must have two RSC cards installed in your Cisco AS5850 system chassis.

Performance Load and Possible Trunk-Card and Port-Density Limitations

The number of CT3, T1, or E1 trunk cards that your system can support depends on the split mode in which it is configured to operate. In classic-split mode, an RSC card needs to handle the trunk cards in its own half only. In handover-split mode, an RSC card needs to be able to handle the full load of trunk cards across the entire chassis. In either case, the number of trunk cards allowed should not exceed the performance load of the handling RSC card.

For further information about performance loads, refer to the tables on Cisco AS5850 universal port capacities in the overview chapter of *Cisco AS5850 Universal Gateway Operations, Administration, Maintenance, and Provisioning Guide*

Throughput Versus Availability

You must choose between maximal throughput and maximal availability:

- Disabling the handover redundancy by configuring classic-split mode provides maximal throughput, at the expense of availability.
- Enabling handover redundancy by configuring handover-split mode provides maximal availability, at the expense of throughput.

Dropped Calls

Calls on a failed RSC, regardless of mode, are lost at the moment of failure.

Fixed Slot Assignments

Slot assignments are fixed and cannot be changed except by a system in handover-split mode during handover. Slots 0-5 are owned by the RSC in slot 6, and slots 8-13 are owned by the RSC in slot 7.

Related Features and Technologies

Router-Shelf Redundancy

The Router-Shelf Redundancy feature that is available on the Cisco AS5800 is similar to RSC Handover Redundancy on the Cisco AS5850.

Related Documents

- *Cisco AS5850 Operations, Administration, Maintenance, and Provisioning Guide*, chapter on provisioning, available from the Cisco AS5850 Product Documentation website

Supported Platforms

- Cisco AS5850 universal gateway

Table 27 Cisco IOS Release and Platform Support for this Feature

Platform	12.2(2)XB1	12.2(11)T
Cisco AS5850	X	X

Determining Platform Support Through Cisco Feature Navigator

Cisco IOS software is packaged in feature sets that support specific platforms. To get updated information regarding platform support for this feature, access Cisco Feature Navigator. Cisco Feature Navigator dynamically updates the list of supported platforms as new platform support is added for the feature.

Cisco Feature Navigator is a web-based tool that enables you to determine which Cisco IOS software images support a specific set of features and which features are supported in a specific Cisco IOS image. You can search by feature or release. Under the release section, you can compare releases side by side to display both the features unique to each software release and the features in common.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/fn>. An account on Cisco.com is not required.

Availability of Cisco IOS Software Images

Platform support for particular Cisco IOS software releases is dependent on the availability of the software images for those platforms. Software images for some platforms may be deferred, delayed, or changed without prior notice. For updated information about platform support and availability of software images for each Cisco IOS software release, refer to the online release notes or, if supported, Cisco Feature Navigator.

Supported Standards MIBs and RFCs

Standards

None

MIBs

- CISCO-RF-MIB

To obtain lists of supported MIBs by platform and Cisco IOS release, and to download MIB modules, go to the Cisco MIB website on Cisco.com at the following URL: <http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml>

RFCs

None

Prerequisites

RSC Cards

Be sure that you have two RSC cards installed in your Cisco AS5850, one in slot 6 and one in slot 7.

Trunk Cards

If you have CT3, T1, or E1 trunk cards in your Cisco AS5850, be sure that you have a supportable number. For more information on performance loads, see the [Restrictions](#), page 363.

Cisco IOS Image

- For classic-split mode, it is advisable, although not mandatory, to configure each RSC with the same Cisco IOS image.
- For handover-split mode, it is mandatory that you configure each RSC with the same Cisco IOS image and the same configuration except for the IP address on egress interfaces. Your Cisco IOS image must support redundancy (Cisco IOS Release 12.2(2)XB, Cisco IOS Release 12.2(11)T, or later releases).

You must replicate the startup configuration for all line cards in the system in both RSCs' saved configurations to ensure correct operation after a handover.

- You can download software configurations to your Cisco AS5850 using Simple Network Management Protocol (SNMP) or a Telnet connection. To learn how to upgrade your Cisco IOS image, go to the Cisco.com website for Cisco AS5850 Product Documentation , locate the *Cisco AS5850 Universal Gateway Operations, Administration, Maintenance, and Provisioning Guide* , and consult the chapter on provisioning.

Configuration Tasks

See the following sections for configuration tasks for this feature. Each task in the list is identified as either required or optional. Note that you must configure and verify either classic-split mode (the default) or handover-split mode.

- [Configuring Classic-Split Mode](#), page 366
- [Verifying Classic-Split Mode](#), page 367
- [Configuring Handover-Split Mode](#), page 368
- [Configuring Handover-Split Mode](#), page 368
- [Verifying Handover-Split Mode](#), page 369
- [Troubleshooting Tips](#), page 372
- [Monitoring and Maintaining Handover Redundancy](#), page 372

Configuring Classic-Split Mode

SUMMARY STEPS

1. Router# **configuration terminal**
2. Router(config)# **redundancy**
3. Router(config-red)# **mode classic-split**
4. Router# **copy running-config startup-config**
5. Router# **reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# configuration terminal	Enters configuration mode.
Step 2	Router(config)# redundancy	Enters configuration-redundancy mode.
Step 3	Router(config-red)# mode classic-split	Selects classic-split (the default) mode.
Step 4	Router# copy running-config startup-config	Copies the running configuration into the startup configuration.
Step 5	Router# reload	Reloads the RSC.

Connect to each RSC in turn and enter these commands.



Note

Classic-split mode is the default mode. If you do not perform these steps, your system defaults to this mode.



Note

These steps simply configure the system to classic-split mode. You must also configure each of the cards manually.

A classic-split system appears to SNMP management applications as two separate Cisco AS5850s. You must conduct a console session for each RSC (two console sessions) to configure your splits. The system controller manages a classic-split configuration as two separate Cisco AS5850 universal gateways.

Network management systems (NMSs) such as the Cisco Universal Gateway Manager (Cisco UGM) are available that provide a single system view of multiple points of presence (POPs) as they monitor performance and log accounting data. An NMS has a graphical user interface (GUI); runs on a UNIX SPARC station; and includes a database-management system, polling engine, trap management, and map integration. The NMS can be installed at a remote facility so that you can access multiple systems through a console port or Web interface.

In classic-split mode, it is desirable--and, with an NMS, essential--to use four unique IDs, one for each RSC and one for each set of slots. In some cases, however, it is sufficient to use the same ID for the two RSCs.

Verifying Classic-Split Mode

In classic-split mode, most **show** commands (with exceptions noted below) display information for only those slots owned by the RSC; they look and behave as they would if there were no cards in the slots that the RSC does not own. To see **show** command information for a slot, you must connect to the RSC that owns that slot.

Enter any of the following commands, in any order.

- To display information about all slots, regardless of ownership, enter the **show context all** command in EXEC mode.
- To display information about owned slots, enter the **show context** command in EXEC mode without the **all** option.
- To display additional relevant output, including whether an RSC is running in classic-split mode and, if so, which slots it owns, enter the **show chassis** command in EXEC mode.

```
RouterA# show chassis
System is in classic-split mode, RSC in slot 6.
Slots owned: 0 1 2 3 4 5
Slots configured: 0 1 2 3 4 5
Slots owned by other: 8 9 10 11 12 13
Slot   Board   CPU      DRAM      I/O Memory  State      Elapsed
      Type   Util    Total (free)  Total (free)
1     UP324   0%/0%  60159040( 51%) 67108864( 73%) Up         6d01h
2     UP324   0%/0%  60159040( 56%) 67108864( 73%) Up         6d01h
3     UP324   0%/0%  60159040( 56%) 67108864( 73%) Up         6d01h
4     CT3_UP216 0%/0%  60159040( 50%) 67108864( 72%) Up         6d01h
System set for auto boot
RouterB# show chassis
System is in classic-split mode, RSC in slot 7.
Slots owned: 8 9 10 11 12 13
Slots configured: 8 9 10 11 12 13
Slots owned by other: 0 1 2 3 4 5
Slot   Board   CPU      DRAM      I/O Memory  State      Elapsed
      Type   Util    Total (free)  Total (free)
9     CT3_UP216 0%/0%  60159040( 65%) 67108864( 72%) Up         00:21:46
10    UP324   0%/0%  60159040( 62%) 67108864( 73%) Up         00:21:48
11    UP324   0%/0%  60159040( 62%) 67108864( 73%) Up         00:21:49
System set for auto boot
```

- To display all configured clock sources, even those from non-owned cards, enter the **show chassis clocks** command in EXEC mode. Only one RSC can provide the master clock, and it may need to have backup clock sources configured from all cards present, regardless of ownership.

```
RouterA# show chassis clocks
Primary Clock:
-----
Slot 6:
System primary is Slot: 4 Port: 1 of priority 10
TDM Bus Master Clock Generator State = NORMAL
Backup clocks:
Source Slot Port DS3-Port Priority Status State
-----
Trunk 9 1 0 8 Good Configured
Trunk 4 21 0 498 Good Default
Trunk 9 21 0 503 Good Default
Status of trunk clocks:
-----
Ds3
Slot Port Type 8 7 6 5 4 3 2 1 0 9 8 7 6 5 4 3 2 1 0 9 8 7 6 5 4 3 2 1
4 0 T3 B B B B B B B G G G G G G G G G G G G G G G G G G G G G
9 0 T3 B B B B B B B G G G G G G G G G G G G G G G G G G G G G
```

Configuring Handover-Split Mode

Perform the following steps on both RSCs so that all cards are configured on both RSCs.

SUMMARY STEPS

1. Router# **configuration terminal**
2. Router(config)# **redundancy**
3. Router(config-red)# **mode handover-split**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# configuration terminal	Enters configuration mode.
Step 2	Router(config)# redundancy	Enters redundancy configuration mode.
Step 3	Router(config-red)# mode handover-split	Selects handover-split mode.

Configuring Handover-Split Mode

Connect to each RSC in turn, change the running configuration so that all cards are configured on this RSC, and perform the following steps.

SUMMARY STEPS

1. Router# **copy running-config startup-config**
2. Router# **dir[/all][filesystem:][file-url]**
3. Router# **reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# copy running-config startup-config	Copies the running configuration into the startup configuration.
Step 2	Router# dir[/all][filesystem:][file-url]	Displays a list of files on a file system. Use to verify that the new image is loaded to system Flash memory or the FTP server.
Step 3	Router# reload	Reloads the RSC.

The net result, when you are done, is that all cards are configured on each RSC.



Note

These steps simply configure the system to handover-split mode. You must also manually configure each card on both RSCs.

**Note**

By default, a single RSC can handle only up to two CT3 cards. You can release this restriction by using the **no dial-config-guidelines** command. For more information on performance loads, see the [Restrictions, page 363](#).

Verifying Handover-Split Mode

Enter any of the following commands, in any order.

- To indicate whether handover is enabled and whether this RSC is active or standby, enter the **show redundancy states** command in EXEC mode.

```
RouterA# show redundancy states
my state = 13 -ACTIVE
peer state = 13 -ACTIVE
Mode = Duplex
Unit = Preferred Primary
Unit ID = 6
Redundancy Mode = Handover-split: If one RSC fails, the peer RSC will take over the
feature boards
Maintenance Mode = Disabled
Manual Swact = Enabled
Communications = Up
client count = 3
client_notification_TMR = 30000 milliseconds
keep_alive TMR = 4000 milliseconds
keep_alive count = 0
keep_alive threshold = 7
RF debug mask = 0x0
```

- To display logged handover event, enter the **show redundancy history** command in EXEC mode.

```
RouterA# show redundancy history
Redundancy Facility Event Log:
00:00:00 client added: RF_INTERNAL_MSG(0) seq=0
00:00:00 client added: RF_LAST_CLIENT(65000) seq=65000
00:00:09 client added: Rsc split dshelf client(19) seq=800
00:00:09 *my state = INITIALIZATION(2) *peer state = DISABLED(1)
00:00:09 RF_PROG_INITIALIZATION(100) RF_INTERNAL_MSG(0) op=0 rc=11
00:00:09 RF_PROG_INITIALIZATION(100) Rsc split dshelf client(19) op=0 rc=11
00:00:09 RF_PROG_INITIALIZATION(100) RF_LAST_CLIENT(65000) op=0 rc=11
00:00:09 *my state = NEGOTIATION(3) peer state = DISABLED(1)
00:00:11 RF_STATUS_PEER_PRESENCE(400) op=1
00:00:11 RF_STATUS_PEER_PRESENCE(400) Rsc split dshelf client(19) op=1
00:00:11 RF_STATUS_PEER_COMM(401) op=1
00:00:11 RF_STATUS_PEER_COMM(401) Rsc split dshelf client(19) op=1
00:00:11 my state = NEGOTIATION(3) *peer state = UNKNOWN(0)
00:00:15 RF_EVENT_CLIENT_PROGRESSION(503) RF_LAST_CLIENT(65000) op=1
00:00:15 RF_PROG_PLATFORM_SYNC(300) RF_LAST_CLIENT(65000) op=1 rc=0
00:00:15 RF_EVENT_CLIENT_PROGRESSION(503) RF_LAST_CLIENT(65000) op=1 rc=0
00:00:17 RF_STATUS_REDUNDANCY_MODE_CHANGE(405) Rsc split dshelf client(19) op=3
00:00:17 RF_EVENT_GO_STANDBY(512) op=0
00:00:17 *my state = STANDBY COLD(4) peer state = UNKNOWN(0)
00:00:17 RF_PROG_STANDBY_COLD(101) RF_INTERNAL_MSG(0) op=0 rc=11
00:00:17 RF_PROG_STANDBY_COLD(101) Rsc split dshelf client(19) op=0 rc=11
00:00:17 RF_PROG_STANDBY_COLD(101) RF_LAST_CLIENT(65000) op=0 rc=11
00:00:19 my state = STANDBY COLD(4) *peer state = ACTIVE_EXTRALOAD(14)
00:00:51 Configuration parsing complete
00:00:53 System initialization complete
00:01:11 RF_STATUS_PEER_PRESENCE(400) op=0
00:01:11 RF_STATUS_PEER_PRESENCE(400) Rsc split dshelf client(19) op=0
00:01:11 my state = STANDBY COLD(4) *peer state = DISABLED(1)
00:01:11 Reloading peer (peer presence lost)
00:01:11 *my state = ACTIVE-FAST(9) peer state = DISABLED(1)
00:01:11 RF_STATUS_MAINTENANCE_ENABLE(403) Rsc split dshelf client(19) op=0
```

```

00:01:11 RF_PROG_ACTIVE_FAST(200) RF_INTERNAL_MSG(0) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_FAST(200) Rsc split dshelf client(19) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_FAST(200) RF_LAST_CLIENT(65000) op=0 rc=11
00:01:11 *my state = ACTIVE-DRAIN(10) peer state = DISABLED(1)
00:01:11 RF_PROG_ACTIVE_DRAIN(201) RF_INTERNAL_MSG(0) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_DRAIN(201) Rsc split dshelf client(19) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_DRAIN(201) RF_LAST_CLIENT(65000) op=0 rc=11
00:01:11 *my state = ACTIVE_PRECONFIG(11) peer state = DISABLED(1)
00:01:11 RF_PROG_ACTIVE_PRECONFIG(202) RF_INTERNAL_MSG(0) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_PRECONFIG(202) Rsc split dshelf client(19) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_PRECONFIG(202) RF_LAST_CLIENT(65000) op=0 rc=11
00:01:11 *my state = ACTIVE_POSTCONFIG(12) peer state = DISABLED(1)
00:01:11 RF_PROG_ACTIVE_POSTCONFIG(203) RF_INTERNAL_MSG(0) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_POSTCONFIG(203) Rsc split dshelf client(19) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_POSTCONFIG(203) RF_LAST_CLIENT(65000) op=0 rc=11
00:01:11 *my state = ACTIVE(13) peer state = DISABLED(1)
00:01:11 RF_PROG_ACTIVE(204) RF_INTERNAL_MSG(0) op=0 rc=11
00:01:11 RF_PROG_ACTIVE(204) Rsc split dshelf client(19) op=0 rc=11
00:01:11 RF_PROG_ACTIVE(204) RF_LAST_CLIENT(65000) op=0 rc=11
00:01:11 RF_STATUS_PEER_COMM(401) op=0
00:01:11 RF_STATUS_PEER_COMM(401) Rsc split dshelf client(19) op=0
00:01:11 Reloading peer (communication down)
00:01:11 RF_EVENT_GO_ACTIVE_EXTRALOAD(513) RF_INTERNAL_MSG(0) op=0
00:01:11 RF_PROG_EXTRALOAD(301) RF_INTERNAL_MSG(0) op=0 rc=11
00:01:11 RF_PROG_EXTRALOAD(301) Rsc split dshelf client(19) op=0 rc=11
00:01:11 RF_PROG_EXTRALOAD(301) RF_LAST_CLIENT(65000) op=0 rc=11
00:01:11 RF_EVENT_GO_ACTIVE_EXTRALOAD(513) RF_INTERNAL_MSG(0) op=0
00:03:02 RF_STATUS_PEER_PRESENCE(400) op=1
00:03:02 RF_STATUS_PEER_PRESENCE(400) Rsc split dshelf client(19) op=1
00:03:02 RF_STATUS_PEER_COMM(401) op=1
00:03:02 RF_STATUS_PEER_COMM(401) Rsc split dshelf client(19) op=1
00:03:02 *my state = ACTIVE_EXTRALOAD(14) *peer state = UNKNOWN(0)
00:03:02 RF_PROG_PLATFORM_SYNC(300) RF_INTERNAL_MSG(0) op=0 rc=11
00:03:02 RF_PROG_PLATFORM_SYNC(300) Rsc split dshelf client(19) op=0 rc=11
00:03:02 RF_PROG_PLATFORM_SYNC(300) RF_LAST_CLIENT(65000) op=0 rc=0
00:03:02 RF_EVENT_CLIENT_PROGRESSION(503) RF_LAST_CLIENT(65000) op=1 rc=0
00:03:02 my state = ACTIVE_EXTRALOAD(14) *peer state = NEGOTIATION(3)
00:03:02 RF_EVENT_PEER_PROG_DONE(506) RF_LAST_CLIENT(65000) op=300
00:03:06 my state = ACTIVE_EXTRALOAD(14) *peer state = STANDBY COLD(4)
6d01h RF_EVENT_GO_ACTIVE_HANDBACK(514) RF_INTERNAL_MSG(0) op=0
6d01h RF_PROG_HANDBACK(302) RF_INTERNAL_MSG(0) op=0 rc=11
6d01h RF_PROG_HANDBACK(302) Rsc split dshelf client(19) op=0 rc=0
6d01h RF_EVENT_CLIENT_PROGRESSION(503) Rsc split dshelf client(19) op=1 rc=0
6d01h RF_EVENT_GO_ACTIVE(511) op=0
6d01h Reloading peer (this unit becoming active)
6d01h *my state = ACTIVE-FAST(9) peer state = STANDBY COLD(4)
6d01h RF_STATUS_MAINTENANCE_ENABLE(403) Rsc split dshelf client(19) op=0
6d01h RF_PROG_ACTIVE_FAST(200) RF_INTERNAL_MSG(0) op=0 rc=11
6d01h RF_PROG_ACTIVE_FAST(200) Rsc split dshelf client(19) op=0 rc=11
6d01h RF_PROG_ACTIVE_FAST(200) RF_LAST_CLIENT(65000) op=0 rc=11
6d01h *my state = ACTIVE-DRAIN(10) peer state = STANDBY COLD(4)
6d01h RF_PROG_ACTIVE_DRAIN(201) RF_INTERNAL_MSG(0) op=0 rc=11
6d01h RF_PROG_ACTIVE_DRAIN(201) Rsc split dshelf client(19) op=0 rc=11
6d01h RF_PROG_ACTIVE_DRAIN(201) RF_LAST_CLIENT(65000) op=0 rc=11
6d01h *my state = ACTIVE_PRECONFIG(11) peer state = STANDBY COLD(4)
6d01h RF_PROG_ACTIVE_PRECONFIG(202) RF_INTERNAL_MSG(0) op=0 rc=11
6d01h RF_PROG_ACTIVE_PRECONFIG(202) Rsc split dshelf client(19) op=0 rc=11
6d01h RF_PROG_ACTIVE_PRECONFIG(202) RF_LAST_CLIENT(65000) op=0 rc=11
6d01h *my state = ACTIVE_POSTCONFIG(12) peer state = STANDBY COLD(4)
6d01h RF_PROG_ACTIVE_POSTCONFIG(203) RF_INTERNAL_MSG(0) op=0 rc=11
6d01h RF_PROG_ACTIVE_POSTCONFIG(203) Rsc split dshelf client(19) op=0 rc=11
6d01h RF_PROG_ACTIVE_POSTCONFIG(203) RF_LAST_CLIENT(65000) op=0 rc=11
6d01h *my state = ACTIVE(13) peer state = STANDBY COLD(4)
6d01h RF_PROG_ACTIVE(204) RF_INTERNAL_MSG(0) op=0 rc=11
6d01h RF_PROG_ACTIVE(204) Rsc split dshelf client(19) op=0 rc=0
6d01h RF_EVENT_CLIENT_PROGRESSION(503) Rsc split dshelf client(19) op=1 rc=0
6d01h my state = ACTIVE(13) *peer state = ACTIVE(13)
6d01h my state = ACTIVE(13) *peer state = UNKNOWN(0)
6d01h Reloading peer (notification timeout)
6d01h my state = ACTIVE(13) *peer state = ACTIVE(13)
6d01h RF_STATUS_REDUNDANCY_MODE_CHANGE(405) Rsc split dshelf client(19) op=1
6d01h RF_EVENT_GO_ACTIVE(511) op=0

```

```
6d01h RF_STATUS_REDUNDANCY_MODE_CHANGE(405) Rsc split dshelf client(19) op=3
6d01h RF_EVENT_GO_ACTIVE(511) op=0
```

- To display details of any pending handover, enter the **show redundancy handover** command in EXEC mode.

```
RouterA# show redundancy handover
No handover pending
```

- To display up to 256 relevant debug entries, enter the **show redundancy debug-log** command in EXEC mode.
- To display additional relevant output, enter the **show chassis** command in EXEC mode. In handover-split mode, this command shows the RSC to be configured with all slots of the entire chassis, regardless of whether the RSC owns the slots or not. Slots owned by the peer RSC are shown to be in the ignore state, properly configured and ready to go.

The following example shows output for two RSCs in normal-load state.

```
RouterA# show chassis
System is in handover-split mode, RSC in slot 6.
  Slots owned: 0 1 2 3 4 5
  Slots configured: 0 1 2 3 4 5 8 9 10 11 12 13
  Slots owned by other: 8 9 10 11 12 13
Slot  Board      CPU      DRAM      I/O Memory  State      Elapsed
   Type      Util      Total (free)  Total (free)
  1    UP324    17%/17%  60159040( 50%) 67108864( 73%) Up         6d01h
  2    UP324     1%/0%   60159040( 56%) 67108864( 73%) Up         6d01h
  3    UP324     0%/0%   60159040( 56%) 67108864( 73%) Up         6d01h
  4    CT3_UP216 1%/0%   60159040( 49%) 67108864( 72%) Up         6d01h
  9    CT3_UP216      60159040( 0%) 67108864( 0%) Ignore    00:00:20
 10    UP324      60159040( 0%) 67108864( 0%) Ignore    00:00:19
 11    UP324      60159040( 0%) 67108864( 0%) Ignore    00:00:18
System set for auto boot
RouterB# show chassis
System is in handover-split mode, RSC in slot 7.
  Slots owned: 8 9 10 11 12 13
  Slots configured: 0 1 2 3 4 5 8 9 10 11 12 13
  Slots owned by other: 0 1 2 3 4 5
Slot  Board      CPU      DRAM      I/O Memory  State      Elapsed
   Type      Util      Total (free)  Total (free)
  1    UP324           0( 0%)       0( 0%) Ignore    00:00:38
  2    UP324           0( 0%)       0( 0%) Ignore    00:00:37
  3    UP324           0( 0%)       0( 0%) Ignore    00:00:36
  4    CT3_UP216      0( 0%)       0( 0%) Ignore    00:00:35
  9    CT3_UP216     0%/0%   60159040( 65%) 67108864( 72%) Up         00:23:14
 10    UP324     0%/0%   60159040( 62%) 67108864( 73%) Up         00:23:16
 11    UP324     0%/0%   60159040( 62%) 67108864( 73%) Up         00:23:17
System set for auto boot
```

The following example shows output for one RSC in extraload state.

```
RouterA# show chassis
System is in handover-split mode, RSC in slot 6.
  Slots owned: 0 1 2 3 4 5 8 9 10 11 12 13
  Slots configured: 0 1 2 3 4 5 8 9 10 11 12 13
  Slots owned by other: none
Slot  Board      CPU      DRAM      I/O Memory  State      Elapsed
   Type      Util      Total (free)  Total (free)
  1    UP324     0%/0%   60159040( 50%) 67108864( 73%) Up         6d02h
  2    UP324     1%/0%   60159040( 56%) 67108864( 73%) Up         6d02h
  3    UP324     0%/0%   60159040( 56%) 67108864( 73%) Up         6d02h
  4    CT3_UP216  6%/5%   60159040( 49%) 67108864( 72%) Up         6d02h
  9    CT3_UP216  5%/4%   60159040( 56%) 67108864( 72%) Up         00:10:29
 10    UP324    20%/20% 60159040( 56%) 67108864( 73%) Up         00:10:30
 11    UP324     0%/0%   60159040( 56%) 67108864( 73%) Up         00:10:30
System set for auto boot
```

Troubleshooting Tips

Command	Purpose
<pre>Router# debug redundancy as5850</pre>	<p>Enables or disables redundancy-related debug options (hardware lines, master RSC, FSM events, mode, RF client). Use to view specific relevant debug options. All debug entries continue to be logged even if you disable an option here, and you can always use the show redundancy debug-log command to view them.</p>

Monitoring and Maintaining Handover Redundancy

Command	Purpose
<pre>Router# redundancy handover cancel peer-resources shelf-resources [busyout-period mins at hh:mm day month year]</pre>	<p>Specifies or cancels handover of slots between RSCs. Use during Cisco IOS image upgrades and to return control of slots to an RSC that failed but is now back in service. Specify handover of slots belonging either to the peer RSC (peer-resources) or to the RSC on which the command is run (shelf-resources). Optionally, specify either or both of the following: length of time for which and exact time at which slots should be busied out before handover.</p> <p>Note The shelf-resources option causes the RSC to reload.</p>



Note

You can detect if an RSC is in extraload with control of the entire chassis resources by observing that the master LED for that RSC is on. You can also detect this state by using the **show redundancy states** command.

The following example shows two instances of handover scheduling, verification, cancellation, and verification of cancellation:

```
RouterA# redundancy handover shelf-resources busyout-period 10 at 16:15 5 Sept 2001
Newly entered handover schedule:
Busyout period at 16:15:00 PST Wed Sep 5 2001 for a duration of 10 minute(s)
Handover pending at 16:25:00 PST Wed Sep 5 2001
Clear calls, handover and reload as specified above?[confirm]
RouterA# show redundancy handover
Busyout period at 16:15:00 PST Wed Sep 5 2001 for a duration of 10 minute(s)
Handover pending at 16:25:00 PST Wed Sep 5 2001
RouterA# redundancy handover cancel
Scheduled handover is cancelled
RSC-Slot6# show redundancy handover
No handover pending
RouterA# redundancy handover peer-resources busyout-period 10 at 16:37 5 Sep 2001
Newly entered handover schedule:
Busyout period at 16:37:00 PST Wed Sep 5 2001 for a duration of 10 minute(s)
Handover pending at 16:47:00 PST Wed Sep 5 2001
```

```

Clear calls and handover as specified above?[confirm]
RouterA# show redundancy handover
Busyout period at 16:37:00 PST Wed Sep 5 2001 for a duration of 10 minute(s)
Handover pending at 16:47:00 PST Wed Sep 5 2001
RouterA# redundancy handover cancel
Scheduled handover is cancelled
RouterA# show redundancy handover
No handover pending

```

Configuration Examples

The following example shows a startup configuration that supports redundancy. Note, in the sections on resource-pool range and controller numbers, that every card in the chassis is configured.

```

RouterA# show startup-config
version 12.2
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
service compress-config
!
hostname RouterA
!
redundancy
 mode handover-split
aaa new-model
!
!
aaa group server tacacs+ redline2
!
aaa group server radius RADIUS-GROUP
 server 172.22.51.9 auth-port 1645 acct-port 1646
!
aaa authentication login CONSOLE none
aaa authentication login VTY none
aaa authentication ppp default group RADIUS-GROUP
aaa authentication ppp RADIUS-LIST group RADIUS-GROUP
aaa authorization exec CONSOLE none
aaa authorization exec RADIUS-LIST group RADIUS-GROUP
aaa authorization network default group RADIUS-GROUP if-authenticated
aaa authorization network RADIUS-LIST group RADIUS-GROUP if-authenticated
aaa accounting network default start-stop group RADIUS-GROUP
aaa nas port extended
aaa session-id common
enable password xxx
!
username RouterB password 0 xxx
username 54006
username 54006_1 password 0 xxx
username RouterA password 0 xxx
username 54006_d_119 password 0 xxx
!
resource-pool enable
!
resource-pool group resource group1
 range port 1/0 1/323
 range port 4/20 4/30
!
resource-pool group resource group2
 range port 9/0 9/215
 range port 10/0 10/120
!
resource-pool group resource digital_group_6
 range limit 207
!
resource-pool group resource digital_group
 range limit 116
!

```

```

resource-pool group resource vpdn_dig
  range limit 92
!
resource-pool profile customer 54006_customer
  limit base-size all
  limit overflow-size 0
  resource group1 speech
  dnis group 54006_dnis
!
resource-pool profile customer 54007_customer
  limit base-size all
  limit overflow-size 0
  resource group2 speech
  dnis group 54007_dnis
!
resource-pool profile customer 54006_customer_sync
  limit base-size all
  limit overflow-size 0
  resource digital_group_6 digital
  dnis group 54006_sync_dnis
!
resource-pool profile customer 54007_sync
  limit base-size all
  limit overflow-size 0
  resource digital_group digital
  dnis group 54007_sync_dnis
!
resource-pool profile customer 54007_sync_vpdn
  limit base-size all
  limit overflow-size 0
  resource vpdn_dig digital
  dnis group 54007_sync_vpdn_dnis
clock timezone PST -7
dial-tdm-clock priority 8 trunk-slot 9 ds3-port 0 port 1
dial-tdm-clock priority 10 trunk-slot 4 ds3-port 0 port 1
spe country tl-default
!
spe link-info poll voice 5
!
ip subnet-zero
ip cef distributed
ip ftp source-interface FastEthernet6/0
ip ftp username root
ip ftp password xxxxxx
no ip domain-lookup
!
vpdn enable
!
vpdn-group 1
  request-dialin
  protocol l2f
  source-ip 30.0.0.1
!
chat-script dial "" "ATZ" OK "ATDT\T" TIMEOUT 60 CONNECT
isdn switch-type primary-5ess
!
controller T3 4/0
  framing c-bit
  cablelength 224
  t1 1-28 controller
!
controller T1 4/0:1
  framing esf
  pri-group timeslots 1-24
!
controller T1 4/0:2
  framing esf
  pri-group timeslots 1-24
!
controller T1 4/0:3
  framing esf
  pri-group timeslots 1-24
!

```

```

.
.
.
controller T1 4/0:28
 shutdown
 framing esf
 pri-group timeslots 1-24
!
controller T3 9/0
 framing c-bit
 cablelength 224
 t1 1-28 controller
!
controller T1 9/0:1
 framing esf
 ds0-group 0 timeslots 1-24 type e&m-fgb dtmf dnis
!
controller T1 9/0:2
 framing esf
 ds0-group 0 timeslots 1-24 type e&m-fgb dtmf dnis
!
controller T1 9/0:3
 framing esf
 ds0-group 0 timeslots 1-24 type e&m-fgb dtmf dnis
!
.
.
.
controller T1 9/0:12
 framing esf
 ds0-group 0 timeslots 1-24 type e&m-fgb dtmf dnis
!
controller T1 9/0:13
 framing esf
 pri-group timeslots 1-24
!
.
.
.
controller T1 9/0:21
 framing esf
 pri-group timeslots 1-24
!
controller T1 9/0:22
 shutdown
 framing esf
 ds0-group 0 timeslots 1-24 type e&m-fgb dtmf dnis
!
.
.
.
controller T1 9/0:28
 shutdown
 framing esf
 ds0-group 0 timeslots 1-24 type e&m-fgb dtmf dnis
!
!
interface Loopback0
 ip address 111.111.111.11 255.255.255.0
 no ip mroute-cache
!
interface Serial4/0:1:23
 no ip address
 encapsulation ppp
 ip mroute-cache
 isdn switch-type primary-5ess
 isdn incoming-voice modem
!
interface Serial4/0:2:23
 no ip address
 encapsulation ppp
 ip mroute-cache

```

```

    isdn switch-type primary-5ess
    isdn incoming-voice modem
    !
interface Serial4/0:3:23
no ip address
encapsulation ppp
ip mroute-cache
isdn switch-type primary-5ess
isdn incoming-voice modem
!
.
.
.
interface Serial4/0:10:23
no ip address
encapsulation ppp
ip mroute-cache
isdn switch-type primary-5ess
isdn incoming-voice modem
!
interface Serial4/0:11:23
no ip address
encapsulation ppp
ip mroute-cache
isdn switch-type primary-5ess
isdn incoming-voice modem
!
interface Serial9/0:21:23
ip unnumbered Loopback0
encapsulation ppp
ip mroute-cache
dialer rotary-group 1
dialer-group 1
isdn switch-type primary-5ess
!
interface Group-Async0
ip unnumbered Loopback0
encapsulation ppp
dialer in-band
dialer idle-timeout 36000 either
dialer string 6003
dialer-group 1
async default routing
async mode dedicated
peer default ip address pool KRAMER
ppp max-bad-auth 3
ppp authentication chap pap callin RADIUS_LIST
ppp chap hostname RouterB
ppp chap password 7 xxxxx
group-range 9/00 11/323
!
interface Group-Async1
ip unnumbered Loopback0
encapsulation ppp
dialer in-band
dialer idle-timeout 36000 either
dialer string 6003
dialer-group 1
async default routing
async mode dedicated
peer default ip address pool KRAMER1
ppp max-bad-auth 3
ppp authentication chap pap callin RADIUS_LIST
ppp chap hostname RouterA
ppp chap password 7 xxxxx
group-range 1/00 4/215
!
interface Dialer0
ip unnumbered Loopback0
encapsulation ppp
dialer in-band
dialer idle-timeout 36000 either
dialer string 6003

```

```

dialer-group 1
peer default ip address pool KRAMER1_d_m
no fair-queue
no cdp enable
ppp authentication chap pap callin RADIUS_LIST
ppp chap hostname RouterA
ppp chap password 7 xxxxxx
ppp multilink
!
interface Dialer1
ip unnumbered Loopback0
encapsulation ppp
dialer in-band
dialer idle-timeout 36000 either
dialer string 6003
dialer-group 1
peer default ip address pool KRAMER_d
no cdp enable
ppp max-bad-auth 3
ppp authentication chap pap callin RADIUS_LIST
ppp chap hostname RouterB
ppp chap password 7 xxxxxx
!
interface Dialer2
ip unnumbered Loopback0
encapsulation ppp
dialer in-band
dialer idle-timeout 36000 either
dialer string 6003
dialer-group 1
peer default ip address pool KRAMER1_d
no fair-queue
no cdp enable
ppp authentication chap pap callin RADIUS_LIST
ppp chap hostname RouterA
ppp chap password 7 xxxxxx
!
interface Dialer5
no ip address
no cdp enable
!
interface Dialer6
no ip address
no cdp enable
!
interface Dialer7
no ip address
no cdp enable
!
.
.
.
interface Dialer26
no ip address
no cdp enable
!
ip local pool KRAMER1 10.6.1.1 10.6.1.108
ip local pool KRAMER1 10.6.2.1 10.6.2.108
ip local pool KRAMER1 10.6.3.1 10.6.3.60
ip local pool KRAMER 10.7.1.1 10.7.1.108
ip local pool KRAMER 10.7.2.1 10.7.2.108
ip local pool KRAMER 10.7.3.1 10.7.3.60
ip local pool KRAMER1_d 10.6.4.1 10.6.4.115
ip local pool KRAMER1_d 10.7.4.1 10.7.4.115
ip local pool KRAMER1_d_m 10.6.4.116 10.6.4.163
ip classless
no ip http server
!
ip radius source-interface FastEthernet6/0
!
dialer dnis group 54006_dnis
number 1002
number 1002100212

```

```

!
dialer dnis group 54007_dnis
  number 38327
!
dialer dnis group 54006_sync_dnis
  number 6666
  number 6600
  number 6666666666
!
dialer dnis group 54007_sync_dnis
  number 7700
  number 7700000000
!
dialer dnis group 54007_sync_vpdn_dnis
  number 7777
  number 7777777777
!
dialer dnis group 54007_vpdn_dnis
  number 38777
dialer-list 1 protocol ip permit
no cdp run
!
tacacs-server host 152.22.51.64
tacacs-server timeout 30
tacacs-server key cisco
snmp-server community public RW
snmp-server enable traps rf
!
radius-server configure-nas
radius-server host 172.22.51.9 auth-port 1645 acct-port 1646 non-standard
radius-server retransmit 3
radius-server attribute nas-port format c
radius-server key lab
call rsvp-sync
!
voice-port 4/0:1:D
!
voice-port 4/0:2:D
!
.
.
.
voice-port 4/0:28:D
!
voice-port 9/0:1:0
!
voice-port 9/0:2:0
!
.
.
.
voice-port 9/0:28:0
!
!
line con 0
  password xxxxxx
  logging synchronous
line aux 0
  logging synchronous
  modem InOut
  transport input all
line vty 0 4
  password xxx
  transport preferred telnet
  transport input telnet
line 1/00 4/215
  modem InOut
  no modem status-poll
  no modem log rs232
  transport preferred none
  transport input all
  autoselect during-login
  autoselect ppp

```

```
line 9/00 9/215
  modem InOut
  no modem status-poll
  no modem log rs232
  transport preferred none
  transport input all
  autoselect during-login
  autoselect ppp
line 10/00 11/323
  modem InOut
  no modem status-poll
  no modem log rs232
  transport preferred none
  transport input all
  autoselect during-login
  autoselect ppp
!
end
```

Command Reference

The following commands are introduced or modified in the feature or features documented in this module. For information about these commands, see the *Cisco IOS Interface and Hardware Component Command Reference* at http://www.cisco.com/en/US/docs/ios/interface/command/reference/ir_book.html. For information about all Cisco IOS commands, go to the Command Lookup Tool at <http://tools.cisco.com/Support/CLILookup> or to the *Cisco IOS Master Commands List*.

- **debug redundancy as5850**
- **mode (redundancy)**
- **redundancy handover**
- **show redundancy (5850)**
- **show chassis**

Glossary

classic-split mode --Mode in which system throughput is maximized because slots are split between two RSCs.

handover --The ability of one part of a system to take over resources that were managed by another part of the system when the latter part fails.

handover-split mode --Mode in which system availability is maximized because an RSC can automatically take control over the slots, cards, and calls of the other RSC, should that other RSC fail.

RSC --route switch controller. The card that provides switch functions, routing, management control, clock control, and egress ports.

service-affecting time --Amount of time during which the system is unable to take new calls or carry the full number of calls.

service interruption --Event during which an in-progress call is dropped, requiring the user to call back.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks.

Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



Route Processor Redundancy Plus (RPR+)

Route Processor Redundancy (RPR) provides an alternative to the High System Availability (HSA) feature. HSA enables a system to reset and use a standby Route Switch Processor (RSP) if the active RSP fails. Using RPR, you can reduce unplanned downtime because RPR enables a quicker switchover between an active and standby RSP if the active RSP experiences a fatal error.

RPR Plus (RPR+) is an enhancement of the RPR feature. RPR+ keeps the Versatile Interface Processors (VIPs) from being reset and reloaded when a switchover occurs between the active and standby RSPs.

- [Finding Feature Information](#), page 381
- [Prerequisites for Route Processor Redundancy Plus \(RPR+\)](#), page 381
- [Restrictions for Route Processor Redundancy Plus \(RPR+\)](#), page 382
- [Information About Route Processor Redundancy Plus \(RPR+\)](#), page 382
- [How to Configure Route Processor Redundancy Plus \(RPR+\)](#), page 383
- [Configuration Examples for Route Processor Redundancy Plus \(RPR+\)](#), page 390
- [Additional References](#), page 391
- [Feature Information for Route Processor Redundancy Plus \(RPR+\)](#), page 393
- [Glossary](#), page 393

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.

Prerequisites for Route Processor Redundancy Plus (RPR+)

RPR and RPR+ require a Cisco 7500 series router loaded with two RSP16s, one RSP16 and one RSP8, two RSP8s, or a combination of RSP2s and RSP4s. If you are using the one RSP16 and one RSP8 combination, you must use the same memory--256 MB--in both RSPs because the secondary RSP must be able to support the primary RSP during a failover.

Restrictions for Route Processor Redundancy Plus (RPR+)

- RSP1s do not support RPR or HSA.
- RPR is supported only on routers that support dual RSPs. Only the Cisco 7507 and Cisco 7513 support dual RSPs.
- RPR+ operates only in a system with VIPs as the line cards. Systems with legacy interface processors default to RPR.
- In RPR+ mode, configuration changes done through Simple Network Management Protocol (SNMP) may not be automatically configured on the standby RSP after a switchover occurs.
- RPR+ does not work on routers configured with MPLS.

Information About Route Processor Redundancy Plus (RPR+)

- [RPR, page 382](#)
- [RPR+, page 382](#)

RPR

Route Processor Redundancy (RPR) provides an alternative to the High System Availability (HSA) feature currently available on Cisco 7500 series routers. HSA enables a system to reset and use a standby Route Switch Processor (RSP) if the active RSP fails.

Using RPR, you can reduce unplanned downtime. RPR enables a quicker switchover between an active and standby RSP if the active RSP experiences a fatal error. When you configure RPR, the standby RSP loads a Cisco IOS image on bootup and initializes itself in standby mode. In the event of a fatal error on the active RSP, the system switches to the standby RSP, which reinitializes itself as the active RSP, reloads all of the line cards, and restarts the system.

RPR+

The RPR+ feature is an enhancement of the RPR feature on Cisco 7500 series routers. RPR+ keeps the VIPs from being reset and reloaded when a switchover occurs between the active and standby RSPs. Because VIPs are not reset and microcode is not reloaded on the VIPs, and the time needed to parse the configuration is eliminated, switchover time is reduced to 30 seconds.

The table below describes the average time for a router to switchover to a standby RSP if the active RSP fails.

Table 28 *Average Switchover Time Comparison Table*

Feature	Time to Immediately Switch a Packet on New RSP After Failover	Expected Overall Time to Have New RSP in New High Availability State After Failover	Notes
HSA	10 minutes	20 minutes	System default.

Feature	Time to Immediately Switch a Packet on New RSP After Failover	Expected Overall Time to Have New RSP in New High Availability State After Failover	Notes
RPR	5 minutes	15 minutes	VIPs and legacy interface processors (IPs) supported.
RPR+	30 seconds	11 minutes	VIPs supported. ¹⁹

**Note**

The table above shows average switchover times. Recovery time will vary depending on the configuration of the router.

In the table above we have noted that RPR+ supports up to two legacy IPs in the router if the **service single-slot-reload-enable** command is configured. By default, the existence of any legacy IPs in the router causes all the line cards to be reloaded during an RPR+ switchover and a message similar to the following to be displayed:

```
%HA-2-MAX_NO_Quiesce: 1 linecard(s) not quiesced exceeds limit of 0, all slots will be reloaded.
```

If the **service single-slot-reload-enable** command is configured, then the NO_Quiesce limit is set to two, allowing two quiesce failures during an RPR+ switchover. When more than two legacy IPs exist in the router, all the line cards are reloaded during an RPR+ switchover, and a message similar to the following is displayed:

```
%HA-2-MAX_NO_Quiesce: 3 linecard(s) not quiesced exceeds limit of 2, all slots will be reloaded.
```

How to Configure Route Processor Redundancy Plus (RPR+)

- [Copying an Image onto Active and Standby RSPs, page 383](#)
- [Setting the Configuration Register Boot Variable, page 385](#)
- [Configuring RPR+, page 387](#)
- [Verifying RPR+, page 389](#)

Copying an Image onto Active and Standby RSPs

Perform this task to use TFTP to copy a high availability Cisco IOS image onto the active and standby RSPs.

Before copying a file to flash memory, you must ensure that there is enough space available in flash memory. Compare the size of the file that you are copying to the amount of available flash memory shown. If the space available is less than the space required by the file that you will copy, the copy process will not continue and an error message similar to the following will be displayed:

```
%Error copying tftp://image@server/tftpboot/file-location/image-name (Not enough space on device) .
```

¹⁹ Legacy IPs default to RPR. To allow RPR+ for VIPs when up to two legacy IPs exist in the router, you must configure the service single-slot-reload-enable command. If you do not enable the service single-slot-reload-enable command or if you have more than two legacy IPs, all the line cards are reloaded.

SUMMARY STEPS

1. **enable**
2. **copy tftp slot *slot-number* :**
3. **copy tftp slaveslot *slot-number* :**

DETAILED STEPS**Step 1****enable**

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Router> enable
```

Step 2**copy tftp slot *slot-number* :**

Use this command to copy a high availability Cisco IOS image onto the flash memory card of the active RSP. The **slot *slot-number*** keyword and argument specify the flash memory card of the active RSP.

Example:

```
Router# copy tftp slot0:
Address or name of remote host []? ip-address
```

Enter the IP address of the TFTP server that contains the new image.

Example:

```
Router# 172.18.2.3
Source filename []? image-name
```

Enter the name of the image file that you are copying to the flash memory card.

Example:

```
Router# rsp-pv-mz
Destination file name? [image-name1
] <Return>
```

Enter the name under which you want the image file to appear at the destination. The destination name is optional. To use the same image name as the source file, press the Enter key.

Example:

```
Accessing tftp://ip-address/...
```

Step 3**copy tftp slaveslot *slot-number* :**

Use this command to copy a high availability Cisco IOS image onto the flash memory card of the standby RSP. The **slaveslot *slot-number*** keyword and argument specify the flash memory card of the standby RSP.

Example:

```
Router# copy tftp slaveslot0:  
Address or name of remote host []? ip-address
```

Enter the IP address of the TFTP server that contains the new image.

Example:

```
Router# 172.18.2.3  
Source filename []? image-name
```

Enter the name of the image file that you are copying to the flash memory card.

Example:

```
Router# rsp-pv-mz  
Destination file name? [image-name1  
> <Return>
```

Enter the name under which you want the image file to appear at the destination. The destination name is optional. To use the same image name as the source file, press the Enter key.

Example:

```
Accessing tftp://ip-address/...
```

- [What to Do Next, page 385](#)

What to Do Next

If you do not want to modify the software configuration register boot field, proceed to the [Configuring RPR+, page 387](#).

Setting the Configuration Register Boot Variable

Perform this optional task to modify the software configuration register boot field to ensure that the system boots the same image as that specified by the **hw-module slot image** command in the [Configuring RPR+, page 387](#).

SUMMARY STEPS

1. **enable**
2. **show version**
3. **configure terminal**
4. **boot system flash slot** *slot-number* : [*image-name*]
5. **config-register** *value*
6. **exit**
7. **reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>show version</p> <p>Example:</p> <pre>Router# show version</pre>	<p>Displays the current configuration register setting at the end of the display.</p>
Step 3	<p>configure terminal</p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 4	<p>boot system flash slot <i>slot-number</i> : [<i>image-name</i>]</p> <p>Example:</p> <pre>Router(config)# boot system flash slot0:rsp-pv-mz</pre>	<p>Specifies the filename of an image stored in flash memory.</p> <ul style="list-style-type: none"> • <i>slot-number</i> :--Specifies the active RSP slot where the flash memory card is located. Valid slot numbers are 0 and 1 for the Cisco 7500 series RSP. • <i>image-name</i> --Specifies the name of the image. It is recommended that you set the boot variable so that the system boots the same image as that specified by the hw-module slot <i>slot-number</i> image file-spec command. See Step 3 of the Configuring RPR+, page 387.
Step 5	<p>config-register <i>value</i></p> <p>Example:</p> <pre>Router(config)# config-register 0x2102</pre>	<p>Modifies the existing configuration register setting to reflect the way in which you want to load a system image.</p> <ul style="list-style-type: none"> • Use the <i>value</i> argument to specify the configuration register setting. Valid values are in the range from 0x0 to 0xFFFF. • In this example, when a reload command is issued, the router automatically boots the image specified in the boot system flash <i>image-name</i> configuration.

Command or Action	Purpose
Step 6 <code>exit</code> Example: Router(config)# <code>exit</code>	Exits global configuration mode and returns to privileged EXEC mode.
Step 7 <code>reload</code> Example: Router# <code>reload</code>	Reboots the router to make your configuration changes take effect.

Examples

The following is sample partial output from the **show version** command; the output displays the current configuration register setting.

```
Router# show version
Cisco IOS Software, C7500 Software (C7500-IPBASE-MZ), Version 12.3(7)T, RELEASE)
TAC Support: http://www.cisco.com/tac
Copyright (c) 1986-2004 by Cisco Systems, Inc.
Compiled Fri 16-Jan-04 18:03 by engineer
ROM: System Bootstrap, Version 12.1(3r)T2, RELEASE SOFTWARE (fc1)
.
.
.
Configuration register is 0x2102
```

Configuring RPR+

Perform this task to configure RPR+.



Note

RPR+ operates only in a system with VIPs as the line cards. Systems with legacy IPs default to RPR mode. Up to two legacy IPs can be supported by RPR+ if the **service single-slot-reload-enable** command is configured. For more details, see the [RPR+, page 382](#).

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hw-module slot *slot-number* image *file-spec***
4. Repeat Step 3 for the standby RSP.
5. **redundancy**
6. **mode { hsa| rpr| rpr-plus }**
7. **exit**
8. **copy system:running-config nvram:startup-config**
9. **hw-module sec-cpu reset**

DETAILED STEPS

Command or Action	Purpose
Step 1 enable Example: <pre>Router> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2 configure terminal Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 3 hw-module slot <i>slot-number</i> image <i>file-spec</i> Example: <pre>Router(config)# hw-module slot 6 image slot0:rsp-pv-mz</pre>	Specifies a high availability Cisco IOS image to run on an active RSP. <ul style="list-style-type: none"> • Use the <i>slot-number</i> argument to specify the RSP slot. • Use the <i>file-spec</i> argument to specify the flash memory card to load the image into and the name of the image. • In this example, the active RSP is loaded in slot 6.
Step 4 Repeat Step 3 for the standby RSP. Example: <pre>Router(config)# hw-module slot 7 image slot0:rsp-pv-mz</pre>	Repeat Step 3 to specify a high availability Cisco IOS image to run on the standby RSP. <ul style="list-style-type: none"> • In this example, the standby RSP is loaded in slot 7.

Command or Action	Purpose
Step 5 redundancy Example: <pre>Router(config)# redundancy</pre>	Enters redundancy configuration mode.
Step 6 mode {hsa rpr rpr-plus} Example: <pre>Router(config-r)# mode rpr-plus</pre>	Configures the redundancy mode. <ul style="list-style-type: none"> Use the rpr-plus keyword to configure the mode as RPR+ on both the active and standby RSPs. If no mode is specified, the default mode is HSA.
Step 7 exit Example: <pre>Router(config-r)# exit</pre>	Exits redundancy 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. Exiting global configuration mode after the redundancy mode has been set to RPR+ will trigger a timer to run for a few seconds, after which the standby RSP resets and reloads.
Step 8 copy system:running-config nvram:startup-config Example: <pre>Router# copy system:running-config nvram:startup-config</pre>	(Optional) Copies the running configuration to the startup configuration to save the RPR+ configuration. <ul style="list-style-type: none"> This command can be run manually immediately after exiting global configuration mode when the redundancy mode is set to RPR+, or it can be run after the standby RSP is reloaded and initialized.
Step 9 hw-module sec-cpu reset Example: <pre>Router# hw-module sec-cpu reset</pre>	(Optional) Resets and reloads the standby RSP with the specified Cisco IOS image and executes the image. <ul style="list-style-type: none"> Although changing the redundancy mode to RPR+ will trigger a reload, using this command may initiate the standby RSP reset a few seconds faster than the automatic reload. <p>Note If you do not specify a Cisco IOS image in Step 3, this command loads and executes the bundled default Cisco IOS standby image. The system then operates in HSA mode.</p>

Verifying RPR+

Perform this task to verify whether RPR+ is configured on the router and to display other redundancy statistics.

SUMMARY STEPS

- enable
- show redundancy

DETAILED STEPS

Step 1 **enable**
Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Router> enable
```

Step 2 **show redundancy**
Use this command to verify what type of redundancy is configured on the router and to display other redundancy information.

Example:

```
Router# show redundancy
Operating mode is rpr-plus
redundancy mode rpr-plus
hw-module slot 2 image disk0:rsp-pv-mz
hw-module slot 3 image disk0:rsp-pv-mz
The system total uptime since last reboot is 5 days, 19 hours 36 minutes.
The system has experienced 27 switchovers.
The system has been active (become master) for 5 days, 15 hours 14 minutes.
Reason for last switchover:User forced.
```

Configuration Examples for Route Processor Redundancy Plus (RPR+)

- [Configuring RPR+ Example, page 390](#)

Configuring RPR+ Example

In the following example, the active RSP is installed in slot 2 and the standby RSP is installed in slot 3 of a Cisco 7507 router.

```
Router# copy tftp slot0:rsp-pv-mz
Router# copy tftp slaveslot0:rsp-pv-mz
Router# configure terminal
Router(config)# hw-module slot 2 image slot0:rsp-pv-mz
Router(config)# hw-module slot 3 image slot0:rsp-pv-mz
Router(config)# redundancy
Router(config-r)# mode rpr-plus
Router(config-r)# end
Router# hw-module sec-cpu reset
Router# show running-config
version 12.3(7)T
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service single-slot-reload-enable
```

```

!
hostname Router
!
boot system rcp://path/to/image/rsp-boot-mz
boot system tftp://path/to/image/rsp-boot-mz
boot bootldr bootflash:rsp-boot-mz
enable password password
!
redundancy
 mode rpr-plus ! Indicates that redundancy mode has been configured for RPR+.
!
hw-module slot 2 image slot0:rsp-pv-mz
hw-module slot 3 image slot0:rsp-pv-mz
ip subnet-zero
ip rcmd remote-username Router
ip cef distributed
ip host iphost 192.168.0.1
mpls traffic-eng auto-bw timers
!
!
controller T3 6/0/0
 clock source line
!
!
interface Ethernet0/0/0
 ip address 10.0.0.1 255.255.0.0
 no ip directed-broadcast
 ip route-cache distributed
 no keepalive
.
.
.
exec-timeout 0 0
 history size 40
 transport preferred none
 transport input none
line aux 0
line vty 0 4
 login
!
end

```

Additional References

Related Documents

Related Topic	Document Title
File management and other configuration commands: complete command syntax, command mode, defaults, usage guidelines, and examples	http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123tcr/123tfr/index.htm Cisco IOS Configuration Fundamentals and Network Management Command Reference
File management and other configuration examples	http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123cgcr/cfun_vcg.htm Cisco IOS Configuration Fundamentals and Network Management Configuration Guide

Related Topic	Document Title
Fast Software Upgrade	http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120limit/120st/120st16/st_rpr7x.htm Route Processor Redundancy and Fast Software Upgrade on Cisco 7500 Series Routers
Single Line Card Reload (SLCR)	http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121newft/121limit/121e/121e5/e5_slcr.htm Cisco 7500 Single Line Card Reload feature document

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

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 Route Processor Redundancy Plus (RPR+)

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 29 Feature Information for Phrase Based on Module Title

Feature Name	Releases	Feature Information
Route Processor Redundancy Plus (RPR+)	12.0(19)ST1	This feature was introduced.
Route Processor Redundancy Plus (RPR+)	12.0(22)S	This feature was integrated into Cisco IOS Release 12.0(22)S.
Route Processor Redundancy Plus (RPR+)	12.2(14)S	This feature was integrated into Cisco IOS Release 12.2(14)S.
Route Processor Redundancy Plus (RPR+)	12.3(7)T	This feature was integrated into Cisco IOS Release 12.3(7)T. The following commands are introduced or modified in the feature: hw-module sec-cpu reset, hw-module slot image, redundancy, redundancy force-switchover, show redundancy (HSA redundancy).

Glossary

Active RSP--The RSP that controls and runs the routing protocols and that presents the system management interface.

HSA --High System Availability. HSA enables a system to reset and use a standby RSP if the active RSP fails.

RPR --Route Processor Redundancy. An alternative to HSA that reduces unplanned downtime.

RPR+ --Route Processor Redundancy Plus. An enhancement to RPR in which the standby RSP is fully initialized. An RPR+ switchover does not involve resetting line cards or reloading line card software for VIPs. Legacy interface processors are reset and reloaded during switchover.

RSP --Route Switch Processor. The Route Processor on the Cisco 7500 series router.

Standby RSP --The RSP that waits ready to take over the functions of the active RSP in the event of unplanned or planned downtime.

**Note**

Refer to *Internetworking Terms and Acronyms* for terms not included in this glossary.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



IPv6 Automatic 6to4 Tunnels

This feature provides support for IPv6 automatic 6to4 tunnels. An automatic 6to4 tunnel allows isolated IPv6 domains to be connected over an IPv4 network to remote IPv6 networks.

- [Finding Feature Information, page 395](#)
- [Information About IPv6 Automatic 6to4 Tunnels, page 395](#)
- [How to Configure IPv6 Automatic 6to4 Tunnels, page 396](#)
- [Configuration Examples for IPv6 Automatic 6to4 Tunnels, page 398](#)
- [Additional References, page 398](#)
- [Feature Information for IPv6 Automatic 6to4 Tunnels, page 399](#)

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.

Information About IPv6 Automatic 6to4 Tunnels

- [Automatic 6to4 Tunnels, page 395](#)

Automatic 6to4 Tunnels

An automatic 6to4 tunnel allows isolated IPv6 domains to be connected over an IPv4 network to remote IPv6 networks. The key difference between automatic 6to4 tunnels and manually configured tunnels is that the tunnel is not point-to-point; it is point-to-multipoint. In automatic 6to4 tunnels, routers are not configured in pairs because they treat the IPv4 infrastructure as a virtual nonbroadcast multiaccess (NBMA) link. The IPv4 address embedded in the IPv6 address is used to find the other end of the automatic tunnel.

An automatic 6to4 tunnel may be configured on a border router in an isolated IPv6 network, which creates a tunnel on a per-packet basis to a border router in another IPv6 network over an IPv4 infrastructure. The tunnel destination is determined by the IPv4 address of the border router extracted from the IPv6 address that starts with the prefix 2002::/16, where the format is 2002:border-router-IPv4-address ::/48. Following the embedded IPv4 address are 16 bits that can be used to number networks within the site. The border

router at each end of a 6to4 tunnel must support both the IPv4 and IPv6 protocol stacks. 6to4 tunnels are configured between border routers or between a border router and a host.

The simplest deployment scenario for 6to4 tunnels is to interconnect multiple IPv6 sites, each of which has at least one connection to a shared IPv4 network. This IPv4 network could be the global Internet or a corporate backbone. The key requirement is that each site have a globally unique IPv4 address; the Cisco software uses this address to construct a globally unique 6to4/48 IPv6 prefix. As with other tunnel mechanisms, appropriate entries in a Domain Name System (DNS) that map between hostnames and IP addresses for both IPv4 and IPv6 allow the applications to choose the required address.

How to Configure IPv6 Automatic 6to4 Tunnels

- [Configuring Automatic 6to4 Tunnels, page 396](#)

Configuring Automatic 6to4 Tunnels

Perform this task to configure automatic 6to4 tunnels.

With 6to4 tunnels, the tunnel destination is determined by the border router IPv4 address, which is concatenated to the prefix 2002::/16 in the format 2002:border-router-IPv4-address::/48. The border router at each end of a 6to4 tunnel must support both the IPv4 and IPv6 protocol stacks.



Note

The configuration of only one IPv4-compatible tunnel and one 6to4 IPv6 tunnel is supported on a router. If you choose to configure both of those tunnel types on the same router, we strongly recommend that they do not share the same tunnel source.

The reason that a 6to4 tunnel and an IPv4-compatible tunnel cannot share an interface is that both of them are NBMA "point-to-multipoint" access links and only the tunnel source can be used to reorder the packets from a multiplexed packet stream into a single packet stream for an incoming interface. So when a packet with an IPv4 protocol type of 41 arrives on an interface, that packet is mapped to an IPv6 tunnel interface based on the IPv4 address. However, if both the 6to4 tunnel and the IPv4-compatible tunnel share the same source interface, the router is not able to determine the IPv6 tunnel interface to which it should assign the incoming packet.

IPv6 manually configured tunnels can share the same source interface because a manual tunnel is a "point-to-point" link, and both the IPv4 source and IPv4 destination of the tunnel are defined.

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *tunnel-number*
4. **ipv6 address** *ipv6-prefix / prefix-length [eui-64]*
5. **tunnel source** {*ip-address*} *interface-type interface-number*}
6. **tunnel mode ipv6ip 6to4**
7. **exit**
8. **ipv6 route** *ipv6-prefix / prefix-length tunnel tunnel-number*

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 <code>enable</code></p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
<p>Step 2 <code>configure terminal</code></p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
<p>Step 3 <code>interface tunnel <i>tunnel-number</i></code></p> <p>Example:</p> <pre>Router(config)# interface tunnel 0</pre>	<p>Specifies a tunnel interface and number, and enters interface configuration mode.</p>
<p>Step 4 <code>ipv6 address <i>ipv6-prefix / prefix-length</i> [eui-64]</code></p> <p>Example:</p> <pre>Router(config-if)# ipv6 address 2002:c0a8:6301:1::1/64</pre>	<p>Specifies the IPv6 address assigned to the interface and enables IPv6 processing on the interface.</p> <ul style="list-style-type: none"> The 32 bits following the initial 2002::/16 prefix correspond to an IPv4 address assigned to the tunnel source.
<p>Step 5 <code>tunnel source {<i>ip-address</i> <i>interface-type interface-number</i>}</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel source ethernet 0</pre>	<p>Specifies the source interface type and number for the tunnel interface.</p> <p>Note The interface type and number specified in the tunnel source command must be configured with an IPv4 address.</p>
<p>Step 6 <code>tunnel mode ipv6ip 6to4</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel mode ipv6ip 6to4</pre>	<p>Specifies an IPv6 overlay tunnel using a 6to4 address.</p>
<p>Step 7 <code>exit</code></p> <p>Example:</p> <pre>Router(config-if)# exit</pre>	<p>Exits interface configuration mode, and returns the router to global configuration mode.</p>

Command or Action	Purpose
<p>Step 8 <code>ipv6 route ipv6-prefix / prefix-length tunnel tunnel-number</code></p> <p>Example:</p> <p>Example:</p> <pre>Router(config)# ipv6 route 2002::/16 tunnel 0</pre>	<p>Configures a static route for the IPv6 6to4 prefix 2002::/16 to the specified tunnel interface.</p> <p>Note When configuring a 6to4 overlay tunnel, you must configure a static route for the IPv6 6to4 prefix 2002::/16 to the 6to4 tunnel interface.</p> <ul style="list-style-type: none"> The tunnel number specified in the ipv6 route command must be the same tunnel number specified in the interface tunnel command.

Configuration Examples for IPv6 Automatic 6to4 Tunnels

- [Example: Configuring 6to4 Tunnels, page 398](#)

Example: Configuring 6to4 Tunnels

The following example configures a 6to4 tunnel on a border router in an isolated IPv6 network. The IPv4 address is 192.168.99.1, which translates to the IPv6 prefix of 2002:c0a8:6301::/48. The IPv6 prefix is subnetted into 2002:c0a8:6301::/64 for the tunnel interface: 2002:c0a8:6301:1::/64 for the first IPv6 network, and 2002:c0a8:6301:2::/64 for the second IPv6 network. The static route ensures that any other traffic for the IPv6 prefix 2002::/16 is directed to tunnel interface 0 for automatic tunneling.

```
interface Ethernet0
description IPv4 uplink
ip address 192.168.99.1 255.255.255.0
!
interface Ethernet1
description IPv6 local network 1
ipv6 address 2002:c0a8:6301:1::1/64
!
interface Ethernet2
description IPv6 local network 2
ipv6 address 2002:c0a8:6301:2::1/64
!
interface Tunnel0
description IPv6 uplink
no ip address
ipv6 address 2002:c0a8:6301::1/64
tunnel source Ethernet 0
tunnel mode ipv6ip 6to4
!
ipv6 route 2002::/16 tunnel 0
```

Additional References

Related Documents

Related Topic	Document Title
IPv6 addressing and connectivity	<i>IPv6 Configuration Guide</i>
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
IPv6 commands	Cisco IOS IPv6 Command Reference
Cisco IOS IPv6 features	Cisco IOS IPv6 Feature Mapping

Standards and RFCs

Standard/RFC	Title
RFCs for IPv6	IPv6 RFCs

MIBs

MIB	MIBs Link
	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

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for IPv6 Automatic 6to4 Tunnels

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 30 **Feature Information for IPv6 Automatic 6to4 Tunnels**

Feature Name	Releases	Feature Information
IPv6 Tunneling: Automatic 6to4 Tunnels	12.0(22)S	An automatic 6to4 tunnel allows isolated IPv6 domains to be connected over an IPv4 network to remote IPv6 networks. The following commands were introduced or modified: tunnel mode ipv6ip , tunnel source .
	12.2(14)S	
	12.2(28)SB	
	12.2(33)SRA	
	12.2(18)SXE	
	12.2(2)T	
	15.0(1)S	

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



IPv6 Automatic IPv4-Compatible Tunnels

This feature provides support for IPv6 automatic IPv4-compatible tunnels. Automatic IPv4-compatible tunnels use IPv4-compatible IPv6 addresses.

- [Finding Feature Information, page 401](#)
- [Information About IPv6 Automatic IPv4-Compatible Tunnels, page 401](#)
- [How to Configure IPv6 Automatic IPv4-Compatible Tunnels, page 404](#)
- [Configuration Examples for IPv6 Automatic IPv4-Compatible Tunnels, page 405](#)
- [Additional References, page 406](#)
- [Feature Information for IPv6 Automatic IPv4-Compatible Tunnels, page 407](#)

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.

Information About IPv6 Automatic IPv4-Compatible Tunnels

- [Overlay Tunnels for IPv6, page 401](#)
- [Automatic IPv4-Compatible IPv6 Tunnels, page 404](#)

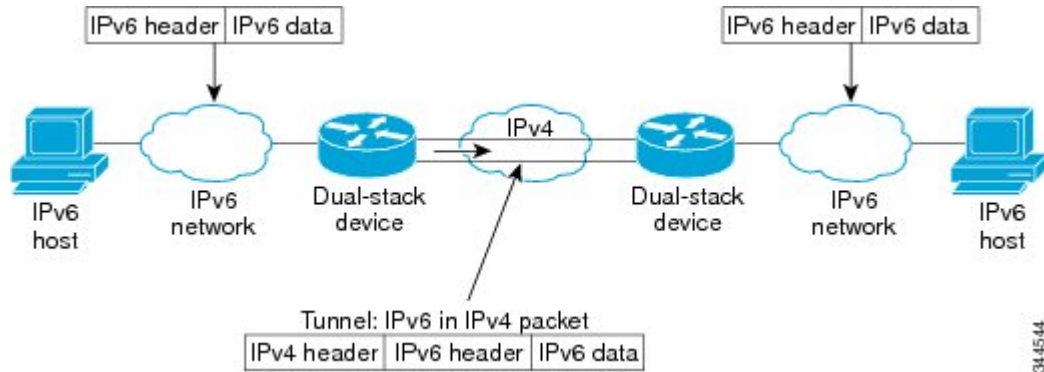
Overlay Tunnels for IPv6

Overlay tunneling encapsulates IPv6 packets in IPv4 packets for delivery across an IPv4 infrastructure (a core network or the figure below). By using overlay tunnels, you can communicate with isolated IPv6 networks without upgrading the IPv4 infrastructure between them. Overlay tunnels can be configured between border devices or between a border device and a host; however, both tunnel endpoints must support both the IPv4 and IPv6 protocol stacks. IPv6 supports the following types of overlay tunneling mechanisms:

- Manual
- Generic routing encapsulation (GRE)
- IPv4-compatible

- 6to4
- Intrasite Automatic Tunnel Addressing Protocol (ISATAP)

Figure 33 **Overlay Tunnels**



Note

Overlay tunnels reduce the maximum transmission unit (MTU) of an interface by 20 octets (assuming that the basic IPv4 packet header does not contain optional fields). A network that uses overlay tunnels is difficult to troubleshoot. Therefore, overlay tunnels that connect isolated IPv6 networks should not be considered a final IPv6 network architecture. The use of overlay tunnels should be considered as a transition technique toward a network that supports both the IPv4 and IPv6 protocol stacks or just the IPv6 protocol stack.

Use the table below to help you determine which type of tunnel that you want to configure to carry IPv6 packets over an IPv4 network.

Table 31 **Suggested Usage of Tunnel Types to Carry IPv6 Packets over an IPv4 Network**

Tunneling Type	Suggested Usage	Usage Notes
Manual	Simple point-to-point tunnels that can be used within a site or between sites.	Can carry IPv6 packets only.
GRE- and IPv4- compatible	Simple point-to-point tunnels that can be used within a site or between sites.	Can carry IPv6, Connectionless Network Service (CLNS), and many other types of packets.
IPv4- compatible	Point-to-multipoint tunnels.	Uses the <code>::/96</code> prefix. We do not recommend using this tunnel type.
6to4	Point-to-multipoint tunnels that can be used to connect isolated IPv6 sites.	Sites use addresses from the <code>2002::/16</code> prefix.

Tunneling Type	Suggested Usage	Usage Notes
6RD	IPv6 service is provided to customers over an IPv4 network by using encapsulation of IPv6 in IPv4.	Prefixes can be from the SP's own address block.
ISATAP	Point-to-multipoint tunnels that can be used to connect systems within a site.	Sites can use any IPv6 unicast addresses.

Individual tunnel types are discussed in detail in this document. We recommend that you review and understand the information about the specific tunnel type that you want to implement. When you are familiar with the type of tunnel you need, see the table below for a summary of the tunnel configuration parameters that you may find useful.

Table 32 Tunnel Configuration Parameters by Tunneling Type

Tunneling Type	Tunnel Configuration Parameter	Tunnel Mode	Tunnel Source	Tunnel Destination	Interface Prefix or Address
Manual	ipv6ip			An IPv4 address, or a reference to an interface on which IPv4 is configured.	An IPv4 address. An IPv6 address.
GRE/IPv4	gre ip				An IPv4 address. An IPv6 address.
IPv4-compatible	ipv6ip auto-tunnel				Not required. These are all point-to-multipoint tunneling types. The IPv4 destination address is calculated, on a per-packet basis, from the IPv6 destination.
6to4	ipv6ip 6to4				Not required. The interface address is generated as <code>::tunnel-source/96</code> . An IPv6 address. The prefix must embed the tunnel source IPv4 address.
6RD	ipv6ip 6rd				An IPv6 address.
ISATAP	ipv6ip isatap				An IPv6 prefix in modified eui-64 format. The IPv6 address is generated from the prefix and the tunnel source IPv4 address.

Automatic IPv4-Compatible IPv6 Tunnels

Automatic IPv4-compatible tunnels use IPv4-compatible IPv6 addresses. IPv4-compatible IPv6 addresses are IPv6 unicast addresses that have zeros in the high-order 96 bits of the address, and an IPv4 address in the low-order 32 bits. They can be written as 0:0:0:0:0:A.B.C.D or ::A.B.C.D, where "A.B.C.D" represents the embedded IPv4 address.

The tunnel destination is automatically determined by the IPv4 address in the low-order 32 bits of IPv4-compatible IPv6 addresses. The host or router at each end of an IPv4-compatible tunnel must support both the IPv4 and IPv6 protocol stacks. IPv4-compatible tunnels can be configured between border-routers or between a border-router and a host. Using IPv4-compatible tunnels is an easy method to create tunnels for IPv6 over IPv4, but the technique does not scale for large networks.

How to Configure IPv6 Automatic IPv4-Compatible Tunnels

- [Configuring IPv4-Compatible IPv6 Tunnels, page 404](#)

Configuring IPv4-Compatible IPv6 Tunnels

Perform this task to configure IPv4-compatible IPv6 tunnels.

With an IPv4-compatible tunnel, the tunnel destination is automatically determined by the IPv4 address in the low-order 32 bits of IPv4-compatible IPv6 addresses. The host or router at each end of an IPv4-compatible tunnel must support both the IPv4 and IPv6 protocol stacks.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *tunnel-number*
4. **tunnel source** {*ip-address*} *interface-type interface-number*}
5. **tunnel mode ipv6ip auto-tunnel**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.

Command or Action	Purpose
<p>Step 3 <code>interface tunnel <i>tunnel-number</i></code></p> <p>Example:</p> <pre>Router(config)# interface tunnel 0</pre>	Specifies a tunnel interface and number, and enters interface configuration mode.
<p>Step 4 <code>tunnel source {<i>ip-address</i> <i>interface-t</i> type <i>interface-number</i>}</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel source ethernet 0</pre>	<p>Specifies the source interface type and number for the tunnel interface.</p> <p>Note The interface type and number specified in the tunnel source command is configured with an IPv4 address only.</p>
<p>Step 5 <code>tunnel mode ipv6ip auto-tunnel</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel mode ipv6ip auto-tunnel</pre>	Specifies an IPv4-compatible tunnel using an IPv4-compatible IPv6 address.

Configuration Examples for IPv6 Automatic IPv4-Compatible Tunnels

- [Example: Configuring IPv4-Compatible IPv6 Tunnels, page 405](#)

Example: Configuring IPv4-Compatible IPv6 Tunnels

The following example configures an IPv4-compatible IPv6 tunnel that allows Border Gateway Protocol (BGP) to run between a number of routers without having to configure a mesh of manual tunnels. Each router has a single IPv4-compatible tunnel, and multiple BGP sessions can run over each tunnel, one to each neighbor. Ethernet interface 0 is used as the tunnel source. The tunnel destination is automatically determined by the IPv4 address in the low-order 32 bits of an IPv4-compatible IPv6 address. Specifically, the IPv6 prefix 0:0:0:0:0 is concatenated to an IPv4 address (in the format 0:0:0:0:0:A.B.C.D or ::A.B.C.D) to create the IPv4-compatible IPv6 address. Ethernet interface 0 is configured with a global IPv6 address and an IPv4 address (the interface supports both the IPv6 and IPv4 protocol stacks).

Multiprotocol BGP is used in the example to exchange IPv6 reachability information with the peer 10.67.0.2. The IPv4 address of Ethernet interface 0 is used in the low-order 32 bits of an IPv4-compatible IPv6 address and is also used as the next-hop attribute. Using an IPv4-compatible IPv6 address for the BGP neighbor allows the IPv6 BGP session to be automatically transported over an IPv4-compatible tunnel.

```
interface tunnel 0
 tunnel source Ethernet 0
 tunnel mode ipv6ip auto-tunnel
interface ethernet 0
 ip address 10.27.0.1 255.255.255.0
 ipv6 address 3000:2222::1/64
```

```

router bgp 65000
  no synchronization
  no bgp default ipv4-unicast
neighbor ::10.67.0.2 remote-as 65002
address-family ipv6
neighbor ::10.67.0.2 activate
neighbor ::10.67.0.2 next-hop-self
network 2001:2222:d00d:b10b::/64

```

Additional References

Related Documents

Related Topic	Document Title
IPv6 addressing and connectivity	<i>IPv6 Configuration Guide</i>
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
IPv6 commands	Cisco IOS IPv6 Command Reference
Cisco IOS IPv6 features	Cisco IOS IPv6 Feature Mapping

Standards and RFCs

Standard/RFC	Title
RFCs for IPv6	IPv6 RFCs

MIBs

MIB	MIBs Link
	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

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for IPv6 Automatic IPv4-Compatible Tunnels

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 33 Feature Information for IPv6 Automatic IPv4-Compatible Tunnels

Feature Name	Releases	Feature Information
IPv6 Tunneling: Automatic IPv4-Compatible Tunnels	12.0(22)S	Automatic IPv4-compatible tunnels use IPv4-compatible IPv6 addresses. The following commands were introduced or modified: tunnel destination , tunnel mode ipv6ip , tunnel source .
	12.2(14)S	
	12.2(28)SB	
	12.2(33)SRA	
	12.2(18)SXE	
	12.2(2)T	
	15.0(1)S	

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams,

and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



IPv6 GRE Tunnels in CLNS Networks

GRE tunneling of IPv4 and IPv6 packets through CLNS networks enables Cisco CTunnels to interoperate with networking equipment from other vendors.

- [Finding Feature Information](#), page 409
- [Information About IPv6 GRE Tunnels in CLNS Networks](#), page 409
- [Configuration Examples for IPv6 GRE Tunnels in CLNS Networks](#), page 412
- [Additional References](#), page 412
- [Feature Information for IPv6 GRE Tunnels in CLNS Networks](#), page 413

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.

Information About IPv6 GRE Tunnels in CLNS Networks

- [Overlay Tunnels for IPv6](#), page 409
- [GRE CLNS Tunnel Support for IPv4 and IPv6 Packets](#), page 411

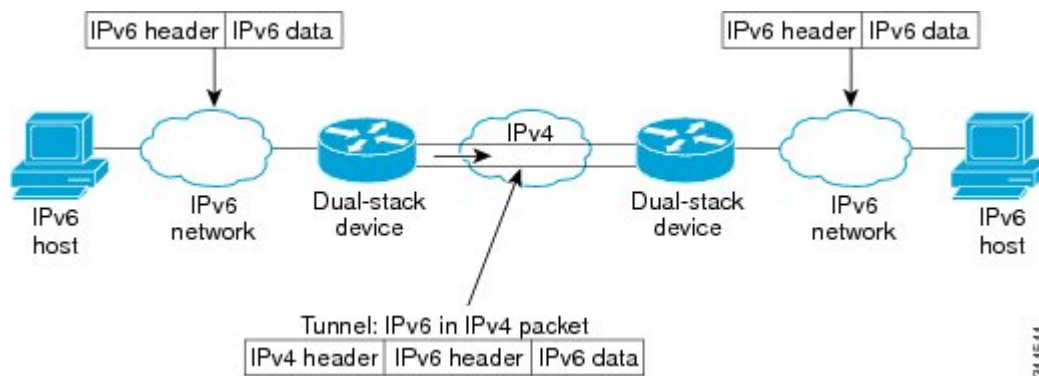
Overlay Tunnels for IPv6

Overlay tunneling encapsulates IPv6 packets in IPv4 packets for delivery across an IPv4 infrastructure (a core network or the figure below). By using overlay tunnels, you can communicate with isolated IPv6 networks without upgrading the IPv4 infrastructure between them. Overlay tunnels can be configured between border devices or between a border device and a host; however, both tunnel endpoints must support both the IPv4 and IPv6 protocol stacks. IPv6 supports the following types of overlay tunneling mechanisms:

- Manual
- Generic routing encapsulation (GRE)
- IPv4-compatible
- 6to4

- Intrasite Automatic Tunnel Addressing Protocol (ISATAP)

Figure 34 **Overlay Tunnels**



Note

Overlay tunnels reduce the maximum transmission unit (MTU) of an interface by 20 octets (assuming that the basic IPv4 packet header does not contain optional fields). A network that uses overlay tunnels is difficult to troubleshoot. Therefore, overlay tunnels that connect isolated IPv6 networks should not be considered a final IPv6 network architecture. The use of overlay tunnels should be considered as a transition technique toward a network that supports both the IPv4 and IPv6 protocol stacks or just the IPv6 protocol stack.

Use the table below to help you determine which type of tunnel that you want to configure to carry IPv6 packets over an IPv4 network.

Table 34 **Suggested Usage of Tunnel Types to Carry IPv6 Packets over an IPv4 Network**

Tunneling Type	Suggested Usage	Usage Notes
Manual	Simple point-to-point tunnels that can be used within a site or between sites.	Can carry IPv6 packets only.
GRE- and IPv4- compatible	Simple point-to-point tunnels that can be used within a site or between sites.	Can carry IPv6, Connectionless Network Service (CLNS), and many other types of packets.
IPv4- compatible	Point-to-multipoint tunnels.	Uses the <code>::/96</code> prefix. We do not recommend using this tunnel type.
6to4	Point-to-multipoint tunnels that can be used to connect isolated IPv6 sites.	Sites use addresses from the <code>2002::/16</code> prefix.
6RD	IPv6 service is provided to customers over an IPv4 network by using encapsulation of IPv6 in IPv4.	Prefixes can be from the SP's own address block.

Tunneling Type	Suggested Usage	Usage Notes
ISATAP	Point-to-multipoint tunnels that can be used to connect systems within a site.	Sites can use any IPv6 unicast addresses.

Individual tunnel types are discussed in detail in this document. We recommend that you review and understand the information about the specific tunnel type that you want to implement. When you are familiar with the type of tunnel you need, see the table below for a summary of the tunnel configuration parameters that you may find useful.

Table 35 Tunnel Configuration Parameters by Tunneling Type

Tunneling Type	Tunnel Configuration Parameter	Tunnel Mode	Tunnel Source	Tunnel Destination	Interface Prefix or Address
Manual	ipv6ip	GRE/IPv4	gre ip	An IPv4 address, or a reference to an interface on which IPv4 is configured.	An IPv4 address.
		IPv4-compatible			Not required. These are all point-to-multipoint tunneling types. The IPv4 destination address is calculated, on a per-packet basis, from the IPv6 destination.
		6to4			An IPv6 address. The prefix must embed the tunnel source IPv4 address.
6RD	ipv6ip 6rd	ISATAP	ipv6ip auto-tunnel	Not required. The interface address is generated as <code>::tunnel-source/96</code> .	An IPv6 address.
					An IPv6 prefix in modified eui-64 format. The IPv6 address is generated from the prefix and the tunnel source IPv4 address.

GRE CLNS Tunnel Support for IPv4 and IPv6 Packets

GRE tunneling of IPv4 and IPv6 packets through CLNS networks enables Cisco CLNS Tunnels (CTunnels) to interoperate with networking equipment from other vendors. This feature provides compliance with RFC 3147.

The optional GRE services defined in header fields, such as checksums, keys, and sequencing, are not supported. Any packet received requesting such services will be dropped.

Configuration Examples for IPv6 GRE Tunnels in CLNS Networks

- [Example: Configuring CTunnels in GRE Mode to Carry IPv6 Packets in CLNS, page 412](#)

Example: Configuring CTunnels in GRE Mode to Carry IPv6 Packets in CLNS

The following example configures a GRE CTunnel running both IS-IS and IPv6 traffic between Router A and Router B in a CLNS network. The **ctunnel mode gre** command allows tunneling between Cisco and third-party networking devices and carries both IPv4 and IPv6 traffic.

The **ctunnel mode gre** command provides a method of tunneling that is compliant with RFC 3147 and allows tunneling between Cisco equipment and third-party networking devices.

Router A

```
ipv6 unicast-routing
clns routing
interface ctunnel 102
  ipv6 address 2001:DB8:1111:2222::1/64
  ctunnel destination 49.0001.2222.2222.2222.00
  ctunnel mode gre
interface Ethernet0/1
  clns router isis
router isis
  net 49.0001.1111.1111.1111.00
```

Router B

```
ipv6 unicast-routing
clns routing
interface ctunnel 201
  ipv6 address 2001:DB8:1111:2222::2/64
  ctunnel destination 49.0001.1111.1111.1111.00
  ctunnel mode gre
interface Ethernet0/1
  clns router isis
router isis
  net 49.0001.2222.2222.2222.00
```

To turn off GRE mode and restore the CTunnel to the default Cisco encapsulation routing only between endpoints on Cisco equipment, use either the **no ctunnel mode** command or the **ctunnel mode cisco** command. The following example shows the same configuration modified to transport only IPv4 traffic.

Additional References

Related Documents

Related Topic	Document Title
IPv6 addressing and connectivity	<i>IPv6 Configuration Guide</i>
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
IPv6 commands	Cisco IOS IPv6 Command Reference
Cisco IOS IPv6 features	Cisco IOS IPv6 Feature Mapping

Standards and RFCs

Standard/RFC	Title
RFCs for IPv6	IPv6 RFCs

MIBs

MIB	MIBs Link
	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

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for IPv6 GRE Tunnels in CLNS Networks

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 36 *Feature Information for IPv6 GRE Tunnels in CLNS Networks*

Feature Name	Releases	Feature Information
CLNS Support for GRE Tunneling of IPv4 and IPv6	12.2(25)S 12.2(33)SRA 12.3(7)T	GRE tunneling of IPv4 and IPv6 packets through CLNS networks enables Cisco CTunnels to interoperate with networking equipment from other vendors. The following commands were introduced or modified: ctunnel mode .

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



ISATAP Tunnel Support for IPv6

ISATAP is an automatic overlay tunneling mechanism that uses the underlying IPv4 network as a NBMA link layer for IPv6

- [Finding Feature Information, page 415](#)
- [Information About ISATAP Tunnel Support for IPv6, page 415](#)
- [How to Configure ISATAP Tunnel Support for IPv6, page 418](#)
- [Configuration Examples for ISATAP Tunnel Support for IPv6, page 420](#)
- [Additional References, page 420](#)
- [Feature Information for ISATAP Tunnel Support for IPv6, page 421](#)

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.

Information About ISATAP Tunnel Support for IPv6

- [Overlay Tunnels for IPv6, page 415](#)
- [ISATAP Tunnels, page 418](#)

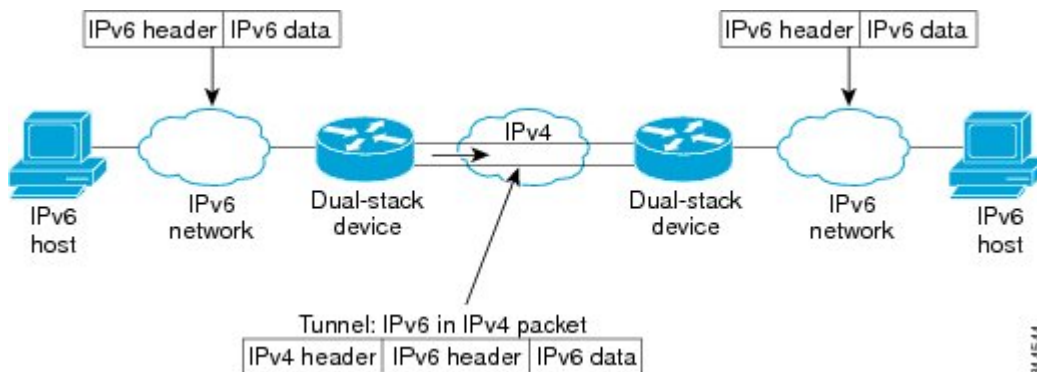
Overlay Tunnels for IPv6

Overlay tunneling encapsulates IPv6 packets in IPv4 packets for delivery across an IPv4 infrastructure (a core network or the figure below). By using overlay tunnels, you can communicate with isolated IPv6 networks without upgrading the IPv4 infrastructure between them. Overlay tunnels can be configured between border devices or between a border device and a host; however, both tunnel endpoints must support both the IPv4 and IPv6 protocol stacks. IPv6 supports the following types of overlay tunneling mechanisms:

- Manual
- Generic routing encapsulation (GRE)
- IPv4-compatible

- 6to4
- Intrasite Automatic Tunnel Addressing Protocol (ISATAP)

Figure 35 **Overlay Tunnels**



Note

Overlay tunnels reduce the maximum transmission unit (MTU) of an interface by 20 octets (assuming that the basic IPv4 packet header does not contain optional fields). A network that uses overlay tunnels is difficult to troubleshoot. Therefore, overlay tunnels that connect isolated IPv6 networks should not be considered a final IPv6 network architecture. The use of overlay tunnels should be considered as a transition technique toward a network that supports both the IPv4 and IPv6 protocol stacks or just the IPv6 protocol stack.

Use the table below to help you determine which type of tunnel that you want to configure to carry IPv6 packets over an IPv4 network.

Table 37 **Suggested Usage of Tunnel Types to Carry IPv6 Packets over an IPv4 Network**

Tunneling Type	Suggested Usage	Usage Notes
Manual	Simple point-to-point tunnels that can be used within a site or between sites.	Can carry IPv6 packets only.
GRE- and IPv4- compatible	Simple point-to-point tunnels that can be used within a site or between sites.	Can carry IPv6, Connectionless Network Service (CLNS), and many other types of packets.
IPv4- compatible	Point-to-multipoint tunnels.	Uses the <code>::/96</code> prefix. We do not recommend using this tunnel type.
6to4	Point-to-multipoint tunnels that can be used to connect isolated IPv6 sites.	Sites use addresses from the <code>2002::/16</code> prefix.

Tunneling Type	Suggested Usage	Usage Notes
6RD	IPv6 service is provided to customers over an IPv4 network by using encapsulation of IPv6 in IPv4.	Prefixes can be from the SP's own address block.
ISATAP	Point-to-multipoint tunnels that can be used to connect systems within a site.	Sites can use any IPv6 unicast addresses.

Individual tunnel types are discussed in detail in this document. We recommend that you review and understand the information about the specific tunnel type that you want to implement. When you are familiar with the type of tunnel you need, see the table below for a summary of the tunnel configuration parameters that you may find useful.

Table 38 Tunnel Configuration Parameters by Tunneling Type

Tunneling Type	Tunnel Configuration Parameter	Tunnel Mode	Tunnel Source	Tunnel Destination	Interface Prefix or Address
Manual	ipv6ip			An IPv4 address, or a reference to an interface on which IPv4 is configured.	An IPv4 address. An IPv6 address.
GRE/IPv4	gre ip				An IPv4 address. An IPv6 address.
IPv4-compatible	ipv6ip auto-tunnel				Not required. These are all point-to-multipoint tunneling types. The IPv4 destination address is calculated, on a per-packet basis, from the IPv6 destination.
6to4	ipv6ip 6to4				Not required. The interface address is generated as <code>::tunnel-source/96</code> . An IPv6 address. The prefix must embed the tunnel source IPv4 address.
6RD	ipv6ip 6rd				An IPv6 address.
ISATAP	ipv6ip isatap				An IPv6 prefix in modified eui-64 format. The IPv6 address is generated from the prefix and the tunnel source IPv4 address.

ISATAP Tunnels

ISATAP is an automatic overlay tunneling mechanism that uses the underlying IPv4 network as an NBMA link layer for IPv6. ISATAP is designed for transporting IPv6 packets within a site where a native IPv6 infrastructure is not yet available; for example, when sparse IPv6 hosts are deployed for testing. ISATAP tunnels allow individual IPv4 or IPv6 dual-stack hosts within a site to communicate with other such hosts on the same virtual link, basically creating an IPv6 network using the IPv4 infrastructure.

The ISATAP router provides standard router advertisement network configuration support for the ISATAP site. This feature allows clients to automatically configure themselves as they would do if they were connected to an Ethernet. It can also be configured to provide connectivity out of the site. ISATAP uses a well-defined IPv6 address format composed of any unicast IPv6 prefix (/64), which can be link local, or global (including 6to4 prefixes), enabling IPv6 routing locally or on the Internet. The IPv4 address is encoded in the last 32 bits of the IPv6 address, enabling automatic IPv6-in-IPv4 tunneling.

Although the ISATAP tunneling mechanism is similar to other automatic tunneling mechanisms, such as IPv6 6to4 tunneling, ISATAP is designed for transporting IPv6 packets within a site, but not between sites.

Although the ISATAP tunneling mechanism is similar to other automatic tunneling mechanisms, such as IPv6 6to4 tunneling, ISATAP is designed for transporting IPv6 packets within a site, not between sites.

ISATAP uses unicast addresses that include a 64-bit IPv6 prefix and a 64-bit interface identifier. The interface identifier is created in modified EUI-64 format in which the first 32 bits contain the value 000:5EFE to indicate that the address is an IPv6 ISATAP address. The table below describes an ISATAP address format.

Table 39 IPv6 ISATAP Address Format

64 Bits	32 Bits	32 Bits
Link local or global IPv6 unicast prefix	0000:5EFE	IPv4 address of the ISATAP link

As shown in the table above, an ISATAP address consists of an IPv6 prefix and the ISATAP interface identifier. This interface identifier includes the IPv4 address of the underlying IPv4 link. The following example shows what an actual ISATAP address would look like if the prefix is 2001:DB8:1234:5678::/64 and the embedded IPv4 address is 10.173.129.8. In the ISATAP address, the IPv4 address is expressed in hexadecimal as 0AAD:8108 (for example, 2001:DB8:1234:5678:0000:5EFE:0AAD:8108).

How to Configure ISATAP Tunnel Support for IPv6

- [Configuring ISATAP Tunnels, page 418](#)

Configuring ISATAP Tunnels

The **tunnel source** command used in the configuration of an ISATAP tunnel must point to an interface with an IPv4 address configured. The ISATAP IPv6 address and prefix (or prefixes) advertised are configured as for a native IPv6 interface. The IPv6 tunnel interface must be configured with a modified EUI-64 address because the last 32 bits in the interface identifier are constructed using the IPv4 tunnel source address.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *tunnel-number*
4. **ipv6 address** *ipv6-prefix / prefix-length* [**eui-64**]
5. **no ipv6 nd ra suppress**
6. **tunnel source** {*ip-address*| *interface-type interface-number*}
7. **tunnel mode ipv6ip isatap**

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 configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3 interface tunnel <i>tunnel-number</i> Example: Router(config)# interface tunnel 1	Specifies a tunnel interface and number, and enters interface configuration mode.
Step 4 ipv6 address <i>ipv6-prefix / prefix-length</i> [eui-64] Example: Router(config-if)# ipv6 address 2001:DB8:6301::/64 eui-64	Specifies the IPv6 address assigned to the interface and enables IPv6 processing on the interface. Note Refer to the <i>Configuring Basic Connectivity for IPv6</i> module for more information on configuring IPv6 addresses.
Step 5 no ipv6 nd ra suppress Example: Router(config-if)# no ipv6 nd ra suppress	Sending of IPv6 router advertisements is disabled by default on tunnel interfaces. This command reenables the sending of IPv6 router advertisements to allow client autoconfiguration.

Command or Action	Purpose
<p>Step 6 <code>tunnel source {ip-address interface-type interface-number}</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel source ethernet 1/0</pre>	<p>Specifies the source interface type and number for the tunnel interface.</p> <p>Note The interface type and number specified in the tunnel source command must be configured with an IPv4 address.</p>
<p>Step 7 <code>tunnel mode ipv6ip isatap</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel mode ipv6ip isatap</pre>	<p>Specifies an IPv6 overlay tunnel using a ISATAP address.</p>

Configuration Examples for ISATAP Tunnel Support for IPv6

- [Example: Configuring ISATAP Tunnels, page 420](#)

Example: Configuring ISATAP Tunnels

The following example shows the tunnel source defined on Ethernet 0 and the **tunnel mode** command used to configure the ISATAP tunnel. Router advertisements are enabled to allow client autoconfiguration.

```
ipv6 unicast-routing
interface tunnel 1
 tunnel source ethernet 0
 tunnel mode ipv6ip isatap
 ipv6 address 2001:DB8::/64 eui-64
 no ipv6 nd ra suppress
 exit
```

Additional References

Related Documents

Related Topic	Document Title
IPv6 addressing and connectivity	<i>IPv6 Configuration Guide</i>
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
IPv6 commands	Cisco IOS IPv6 Command Reference

Related Topic	Document Title
Cisco IOS IPv6 features	Cisco IOS IPv6 Feature Mapping

Standards and RFCs

Standard/RFC	Title
RFCs for IPv6	IPv6 RFCs

MIBs

MIB	MIBs Link
	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

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for ISATAP Tunnel Support for IPv6

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 40 **Feature Information for ISATAP Tunnel Support for IPv6**

Feature Name	Releases	Feature Information
ISATAP Tunnel Support for IPv6	12.2(14)S 12.2(28)SB 12.2(33)SRA 12.2(17a)SXI 12.2(15)T	ISATAP is an automatic overlay tunneling mechanism that uses the underlying IPv4 network as a NBMA link layer for IPv6. The following commands were introduced or modified: ipv6 nd ra suppress , tunnel mode ipv6ip , tunnel source .

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



Manually Configured IPv6 over IPv4 Tunnels

This feature provides support for manually configured IPv6 over IPv4 tunnels. A manually configured tunnel is equivalent to a permanent link between two IPv6 domains over an IPv4 backbone.

- [Finding Feature Information, page 423](#)
- [Information About Manually Configured IPv6 over IPv4 Tunnels, page 423](#)
- [How to Enable Manually Configured IPv6 over IPv4 Tunnels, page 426](#)
- [Configuration Examples for Manually Configured IPv6 over IPv4 Tunnels, page 427](#)
- [Additional References, page 428](#)
- [Feature Information for Manually Configured IPv6 over IPv4 Tunnels, page 429](#)

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.

Information About Manually Configured IPv6 over IPv4 Tunnels

- [Overlay Tunnels for IPv6, page 423](#)
- [IPv6 Manually Configured Tunnels, page 426](#)

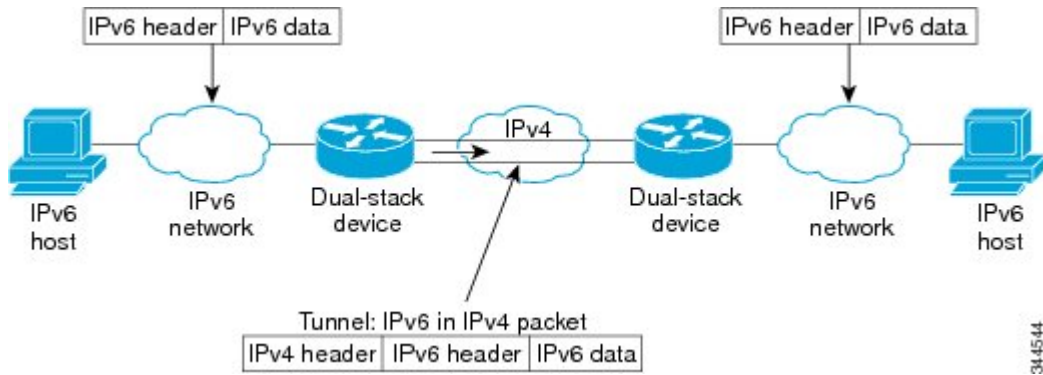
Overlay Tunnels for IPv6

Overlay tunneling encapsulates IPv6 packets in IPv4 packets for delivery across an IPv4 infrastructure (a core network or the figure below). By using overlay tunnels, you can communicate with isolated IPv6 networks without upgrading the IPv4 infrastructure between them. Overlay tunnels can be configured between border devices or between a border device and a host; however, both tunnel endpoints must support both the IPv4 and IPv6 protocol stacks. IPv6 supports the following types of overlay tunneling mechanisms:

- Manual

- Generic routing encapsulation (GRE)
- IPv4-compatible
- 6to4
- Intrasite Automatic Tunnel Addressing Protocol (ISATAP)

Figure 36 *Overlay Tunnels*



Note

Overlay tunnels reduce the maximum transmission unit (MTU) of an interface by 20 octets (assuming that the basic IPv4 packet header does not contain optional fields). A network that uses overlay tunnels is difficult to troubleshoot. Therefore, overlay tunnels that connect isolated IPv6 networks should not be considered a final IPv6 network architecture. The use of overlay tunnels should be considered as a transition technique toward a network that supports both the IPv4 and IPv6 protocol stacks or just the IPv6 protocol stack.

Use the table below to help you determine which type of tunnel that you want to configure to carry IPv6 packets over an IPv4 network.

Table 41 *Suggested Usage of Tunnel Types to Carry IPv6 Packets over an IPv4 Network*

Tunneling Type	Suggested Usage	Usage Notes
Manual	Simple point-to-point tunnels that can be used within a site or between sites.	Can carry IPv6 packets only.
GRE- and IPv4- compatible	Simple point-to-point tunnels that can be used within a site or between sites.	Can carry IPv6, Connectionless Network Service (CLNS), and many other types of packets.
IPv4- compatible	Point-to-multipoint tunnels.	Uses the ::/96 prefix. We do not recommend using this tunnel type.
6to4	Point-to-multipoint tunnels that can be used to connect isolated IPv6 sites.	Sites use addresses from the 2002::/16 prefix.

Tunneling Type	Suggested Usage	Usage Notes
6RD	IPv6 service is provided to customers over an IPv4 network by using encapsulation of IPv6 in IPv4.	Prefixes can be from the SP's own address block.
ISATAP	Point-to-multipoint tunnels that can be used to connect systems within a site.	Sites can use any IPv6 unicast addresses.

Individual tunnel types are discussed in detail in this document. We recommend that you review and understand the information about the specific tunnel type that you want to implement. When you are familiar with the type of tunnel you need, see the table below for a summary of the tunnel configuration parameters that you may find useful.

Table 42 Tunnel Configuration Parameters by Tunneling Type

Tunneling Type	Tunnel Configuration Parameter	Tunnel Mode	Tunnel Source	Tunnel Destination	Interface Prefix or Address
Manual	ipv6ip			An IPv4 address, or a reference to an interface on which IPv4 is configured.	An IPv4 address. An IPv6 address.
GRE/IPv4	gre ip				An IPv4 address. An IPv6 address.
IPv4-compatible	ipv6ip auto-tunnel				Not required. These are all point-to-multipoint tunneling types. The IPv4 destination address is calculated, on a per-packet basis, from the IPv6 destination.
6to4	ipv6ip 6to4				Not required. The interface address is generated as <code>::tunnel-source/96</code> . An IPv6 address. The prefix must embed the tunnel source IPv4 address.
6RD	ipv6ip 6rd				An IPv6 address.
ISATAP	ipv6ip isatap				An IPv6 prefix in modified eui-64 format. The IPv6 address is generated from the prefix and the tunnel source IPv4 address.

IPv6 Manually Configured Tunnels

A manually configured tunnel is equivalent to a permanent link between two IPv6 domains over an IPv4 backbone. The primary use is for stable connections that require regular secure communication between two edge devices or between an end system and an edge device, or for connection to remote IPv6 networks.

An IPv6 address is manually configured on a tunnel interface, and manually configured IPv4 addresses are assigned to the tunnel source and the tunnel destination. The host or device at each end of a configured tunnel must support both the IPv4 and IPv6 protocol stacks. Manually configured tunnels can be configured between border devices or between a border device and a host. Cisco Express Forwarding switching can be used for IPv6 manually configured tunnels, or Cisco Express Forwarding switching can be disabled if process switching is needed.

How to Enable Manually Configured IPv6 over IPv4 Tunnels

- [Configuring Manual IPv6 Tunnels, page 426](#)

Configuring Manual IPv6 Tunnels

Perform this task to configure manual IPv6 tunnels.

With manually configured IPv6 tunnels, an IPv6 address is configured on a tunnel interface, and manually configured IPv4 addresses are assigned to the tunnel source and the tunnel destination. The host or router at each end of a configured tunnel must support both the IPv4 and IPv6 protocol stacks.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *tunnel-number*
4. **ipv6 address** *ipv6-prefix / prefix-length* [**oui-64**]
5. **tunnel source** {*ip-address*| *interface-type interface-number*}
6. **tunnel destination** *ip-address*
7. **tunnel mode ipv6ip**

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.

Command or Action	Purpose
<p>Step 2 <code>configure terminal</code></p> <p>Example:</p> <pre>Router# configure terminal</pre>	Enters global configuration mode.
<p>Step 3 <code>interface tunnel tunnel-number</code></p> <p>Example:</p> <pre>Router(config)# interface tunnel 0</pre>	Specifies a tunnel interface and number, and enters interface configuration mode.
<p>Step 4 <code>ipv6 address ipv6-prefix / prefix-length [eui-64]</code></p> <p>Example:</p> <pre>Router(config-if)# ipv6 address 3ffe:b00:c18:1::3/127</pre>	Specifies the IPv6 network assigned to the interface and enables IPv6 processing on the interface.
<p>Step 5 <code>tunnel source {ip-address interface-type interface-number}</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel source ethernet 0</pre>	<p>Specifies the source IPv4 address or the source interface type and number for the tunnel interface.</p> <ul style="list-style-type: none"> If an interface is specified, the interface must be configured with an IPv4 address.
<p>Step 6 <code>tunnel destination ip-address</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel destination 192.168.30.1</pre>	Specifies the destination IPv4 address or hostname for the tunnel interface.
<p>Step 7 <code>tunnel mode ipv6ip</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel mode ipv6ip</pre>	<p>Specifies a manual IPv6 tunnel.</p> <p>Note The <code>tunnel mode ipv6ip</code> command specifies IPv6 as the passenger protocol and IPv4 as both the encapsulation and transport protocol for the manual IPv6 tunnel.</p>

Configuration Examples for Manually Configured IPv6 over IPv4 Tunnels

- [Example: Configuring Manual IPv6 Tunnels, page 428](#)

Example: Configuring Manual IPv6 Tunnels

The following example configures a manual IPv6 tunnel between Router A and Router B. In the example, tunnel interface 0 for both Router A and Router B is manually configured with a global IPv6 address. The tunnel source and destination addresses are also manually configured.

Router A Configuration

```
interface ethernet 0
 ip address 192.168.99.1 255.255.255.0
interface tunnel 0
 ipv6 address 3ffe:b00:c18:1::3/127
 tunnel source ethernet 0
 tunnel destination 192.168.30.1
 tunnel mode ipv6ip
```

Router B Configuration

```
interface ethernet 0
 ip address 192.168.30.1 255.255.255.0
interface tunnel 0
 ipv6 address 3ffe:b00:c18:1::2/127
 tunnel source ethernet 0
 tunnel destination 192.168.99.1
 tunnel mode ipv6ip
```

Additional References

Related Documents

Related Topic	Document Title
IPv6 addressing and connectivity	IPv6 Configuration Guide
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
IPv6 commands	Cisco IOS IPv6 Command Reference
Cisco IOS IPv6 features	Cisco IOS IPv6 Feature Mapping

Standards and RFCs

Standard/RFC	Title
RFCs for IPv6	IPv6 RFCs

MIBs

MIB	MIBs Link
	<p>To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:</p> <p>http://www.cisco.com/go/mibs</p>

Technical Assistance

Description	Link
<p>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/cisco/web/support/index.html</p>

Feature Information for Manually Configured IPv6 over IPv4 Tunnels

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 43 Feature Information for Manually Configured IPv6 over IPv4 Tunnels

Feature Name	Releases	Feature Information
IPv6 Tunneling: Manually Configured IPv6 over IPv4 Tunnels	12.0(23)S	<p>A manually configured tunnel is equivalent to a permanent link between two IPv6 domains over an IPv4 backbone.</p> <p>The following commands were introduced or modified: tunnel destination, tunnel ipv6ip, tunnel source.</p>
	12.2(14)S	
	12.2(28)SB	
	12.2(33)SRA	
	12.2(2)T	
	15.0(1)S	

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.



IPv6 over IPv4 GRE Tunnels

GRE tunnels are links between two points, with a separate tunnel for each link. The tunnels are not tied to a specific passenger or transport protocol, but in this case carry IPv6 as the passenger protocol with the GRE as the carrier protocol and IPv4 or IPv6 as the transport protocol.

- [Finding Feature Information, page 431](#)
- [Information About IPv6 over IPv4 GRE Tunnels, page 431](#)
- [How to Configure IPv6 over IPv4 GRE Tunnels, page 434](#)
- [Configuration Examples for IPv6 over IPv4 GRE Tunnels, page 435](#)
- [Additional References, page 437](#)
- [Feature Information for IPv6 over IPv4 GRE Tunnels, page 438](#)

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.

Information About IPv6 over IPv4 GRE Tunnels

- [Overlay Tunnels for IPv6, page 431](#)
- [GRE IPv4 Tunnel Support for IPv6 Traffic, page 434](#)

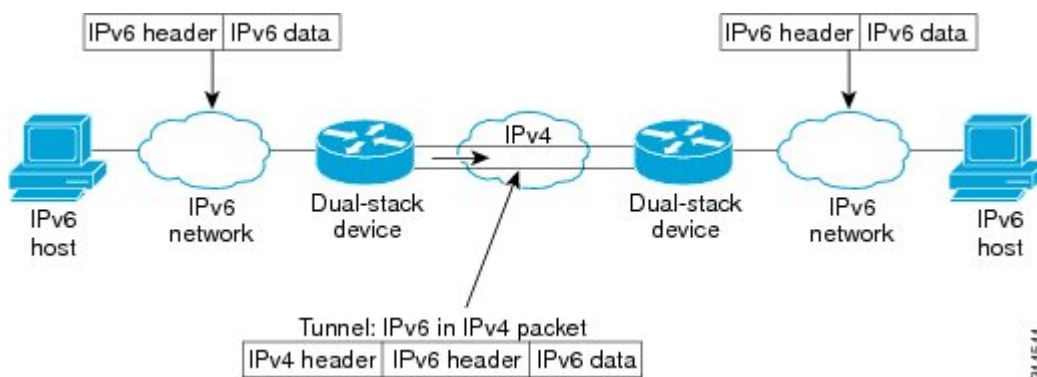
Overlay Tunnels for IPv6

Overlay tunneling encapsulates IPv6 packets in IPv4 packets for delivery across an IPv4 infrastructure (a core network or the figure below). By using overlay tunnels, you can communicate with isolated IPv6 networks without upgrading the IPv4 infrastructure between them. Overlay tunnels can be configured between border devices or between a border device and a host; however, both tunnel endpoints must support both the IPv4 and IPv6 protocol stacks. IPv6 supports the following types of overlay tunneling mechanisms:

- Manual
- Generic routing encapsulation (GRE)

- IPv4-compatible
- 6to4
- Intrasite Automatic Tunnel Addressing Protocol (ISATAP)

Figure 37 **Overlay Tunnels**



Note

Overlay tunnels reduce the maximum transmission unit (MTU) of an interface by 20 octets (assuming that the basic IPv4 packet header does not contain optional fields). A network that uses overlay tunnels is difficult to troubleshoot. Therefore, overlay tunnels that connect isolated IPv6 networks should not be considered a final IPv6 network architecture. The use of overlay tunnels should be considered as a transition technique toward a network that supports both the IPv4 and IPv6 protocol stacks or just the IPv6 protocol stack.

Use the table below to help you determine which type of tunnel that you want to configure to carry IPv6 packets over an IPv4 network.

Table 44 **Suggested Usage of Tunnel Types to Carry IPv6 Packets over an IPv4 Network**

Tunneling Type	Suggested Usage	Usage Notes
Manual	Simple point-to-point tunnels that can be used within a site or between sites.	Can carry IPv6 packets only.
GRE- and IPv4- compatible	Simple point-to-point tunnels that can be used within a site or between sites.	Can carry IPv6, Connectionless Network Service (CLNS), and many other types of packets.
IPv4- compatible	Point-to-multipoint tunnels.	Uses the <code>::/96</code> prefix. We do not recommend using this tunnel type.
6to4	Point-to-multipoint tunnels that can be used to connect isolated IPv6 sites.	Sites use addresses from the <code>2002::/16</code> prefix.

Tunneling Type	Suggested Usage	Usage Notes
6RD	IPv6 service is provided to customers over an IPv4 network by using encapsulation of IPv6 in IPv4.	Prefixes can be from the SP's own address block.
ISATAP	Point-to-multipoint tunnels that can be used to connect systems within a site.	Sites can use any IPv6 unicast addresses.

Individual tunnel types are discussed in detail in this document. We recommend that you review and understand the information about the specific tunnel type that you want to implement. When you are familiar with the type of tunnel you need, see the table below for a summary of the tunnel configuration parameters that you may find useful.

Table 45 Tunnel Configuration Parameters by Tunneling Type

Tunneling Type	Tunnel Configuration Parameter	Tunnel Mode	Tunnel Source	Tunnel Destination	Interface Prefix or Address
Manual	ipv6ip			An IPv4 address, or a reference to an interface on which IPv4 is configured.	An IPv4 address. An IPv6 address.
GRE/IPv4	gre ip				An IPv4 address. An IPv6 address.
IPv4-compatible	ipv6ip auto-tunnel				Not required. These are all point-to-multipoint tunneling types. The IPv4 destination address is calculated, on a per-packet basis, from the IPv6 destination.
6to4	ipv6ip 6to4				Not required. The interface address is generated as <code>::tunnel-source/96</code> . An IPv6 address. The prefix must embed the tunnel source IPv4 address.
6RD	ipv6ip 6rd				An IPv6 address.
ISATAP	ipv6ip isatap				An IPv6 prefix in modified eui-64 format. The IPv6 address is generated from the prefix and the tunnel source IPv4 address.

GRE IPv4 Tunnel Support for IPv6 Traffic

IPv6 traffic can be carried over IPv4 GRE tunnels using the standard GRE tunneling technique that is designed to provide the services to implement any standard point-to-point encapsulation scheme. As in IPv6 manually configured tunnels, GRE tunnels are links between two points, with a separate tunnel for each link. The tunnels are not tied to a specific passenger or transport protocol but, in this case, carry IPv6 as the passenger protocol with the GRE as the carrier protocol and IPv4 or IPv6 as the transport protocol.

The primary use of GRE tunnels is for stable connections that require regular secure communication between two edge devices or between an edge device and an end system. The edge devices and the end systems must be dual-stack implementations.

How to Configure IPv6 over IPv4 GRE Tunnels

- [Configuring GRE IPv6 Tunnels, page 434](#)

Configuring GRE IPv6 Tunnels

Perform this task to configure a GRE tunnel on an IPv6 network. GRE tunnels can be configured to run over an IPv6 network layer and to transport IPv6 packets in IPv6 tunnels and IPv4 packets in IPv6 tunnels.

When GRE IPv6 tunnels are configured, IPv6 addresses are assigned to the tunnel source and the tunnel destination. The tunnel interface can have either IPv4 or IPv6 addresses assigned (this is not shown in the task). The host or router at each end of a configured tunnel must support both the IPv4 and IPv6 protocol stacks.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *tunnel-number*
4. **ipv6 address** *ipv6-prefix / prefix-length* [**oui-64**]
5. **tunnel source** {*ip-address* | *ipv6-address* | *interface-type interface-number*}
6. **tunnel destination** {*host-name* | *ip-address* | *ipv6-address*}
7. **tunnel mode** {**aurp** | **cayman** | **dvmrp** | **eon** | **gre** | **gre multipoint** | **gre ipv6** | **ipip** [**decapsulate-any**] | **iptalk** | **ipv6** | **mpls** | **nos**}

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.

Command or Action	Purpose
<p>Step 2 <code>configure terminal</code></p> <p>Example:</p> <pre>Router# configure terminal</pre>	Enters global configuration mode.
<p>Step 3 <code>interface tunnel tunnel-number</code></p> <p>Example:</p> <pre>Router(config)# interface tunnel 0</pre>	Specifies a tunnel interface and number, and enters interface configuration mode.
<p>Step 4 <code>ipv6 address ipv6-prefix / prefix-length [eui-64]</code></p> <p>Example:</p> <pre>Router(config-if)# ipv6 address 3ffe:b00:c18:1::3/127</pre>	Specifies the IPv6 network assigned to the interface and enables IPv6 processing on the interface.
<p>Step 5 <code>tunnel source {ip-address ipv6-address interface-type interface-number}</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel source ethernet 0</pre>	<p>Specifies the source IPv4 address or the source interface type and number for the tunnel interface.</p> <ul style="list-style-type: none"> If an interface is specified, the interface must be configured with an IPv4 address.
<p>Step 6 <code>tunnel destination {host-name ip-address ipv6-address}</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel destination 2001:DB8:1111:2222::1/64</pre>	Specifies the destination IPv6 address or hostname for the tunnel interface.
<p>Step 7 <code>tunnel mode {aurp cayman dvmrp eon gre gre multipoint gre ipv6 ipip [decapsulate-any] iptalk ipv6 mpls nos}</code></p> <p>Example:</p> <pre>Router(config-if)# tunnel mode gre ipv6</pre>	<p>Specifies a GRE IPv6 tunnel.</p> <p>Note The <code>tunnel mode gre ipv6</code> command specifies GRE as the encapsulation protocol for the tunnel.</p>

Configuration Examples for IPv6 over IPv4 GRE Tunnels

- [Example: GRE Tunnel Running IS-IS and IPv6 Traffic, page 436](#)
- [Example: Tunnel Destination Address for IPv6 Tunnel, page 436](#)

Example: GRE Tunnel Running IS-IS and IPv6 Traffic

The following example configures a GRE tunnel running both IS-IS and IPv6 traffic between Router A and Router B:

Router A Configuration

```

ipv6 unicast-routing
clns routing
!
interface tunnel 0
no ip address
ipv6 address 3ffe:b00:c18:1::3/127
ipv6 router isis
tunnel source Ethernet 0/0
tunnel destination 2001:DB8:1111:2222::1/64
tunnel mode gre ipv6
!
interface Ethernet0/0
ip address 10.0.0.1 255.255.255.0
!
router isis
net 49.0000.0000.000a.00

```

Router B Configuration

```

ipv6 unicast-routing
clns routing
!
interface tunnel 0
no ip address
ipv6 address 3ffe:b00:c18:1::2/127
ipv6 router isis
tunnel source Ethernet 0/0
tunnel destination 2001:DB8:1111:2222::2/64
tunnel mode gre ipv6
!
interface Ethernet0/0
ip address 10.0.0.2 255.255.255.0
!
router isis
net 49.0000.0000.000b.00
address-family ipv6
redistribute static
exit-address-family

```

Example: Tunnel Destination Address for IPv6 Tunnel

```

Router(config
)
# interface Tunnel0
Router(config
-if)
# no ip address
Router(config
-if)
# ipv6 router isis
Router(config
-if)
# tunnel source Ethernet 0/0
Router(config
-if)
# tunnel destination 2001:DB8:1111:2222::1/64
Router(config
-if)

```

```

# tunnel mode gre ipv6
Router(config
-if)
# exit
!
Router(config
)
# interface Ethernet0/0
Router(config
-if)
# ip address 10.0.0.1 255.255.255.0
Router(config
-if)
# exit
!
Router(config
)
# ipv6 unicast-routing
Router(config
)
# router isis

Router(config
)
# net 49.0000.0000.000a.00

```

Additional References

Related Documents

Related Topic	Document Title
IPv6 addressing and connectivity	<i>IPv6 Configuration Guide</i>
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
IPv6 commands	Cisco IOS IPv6 Command Reference
Cisco IOS IPv6 features	Cisco IOS IPv6 Feature Mapping

Standards and RFCs

Standard/RFC	Title
RFCs for IPv6	IPv6 RFCs

MIBs

MIB	MIBs Link
	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

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for IPv6 over IPv4 GRE Tunnels

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 46 *Feature Information for IPv6 over IPv4 GRE Tunnels*

Feature Name	Releases	Feature Information
IPv6 over IPv4 GRE Tunnels	12.0(22)S	GRE tunnels are links between two points, with a separate tunnel for each link. The tunnels are not tied to a specific passenger or transport protocol, but in this case carry IPv6 as the passenger protocol with the GRE as the carrier protocol and IPv4 or IPv6 as the transport protocol.
	12.2(14)S	
	12.2(28)SB	
	12.2(33)SRA	
	12.2(17a)SX1	
	12.2(4)T	
	12.3	
	12.3(2)T	
	12.4	
	12.4(2)T	
	15.0(1)S	

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks.

Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

