

Steelhead® Appliance Deployment Guide

Including the Steelhead® Mobile Controller

April 2014



© 2014 Riverbed Technology. All rights reserved.

Riverbed®, Cloud Steelhead®, Granite™, Interceptor®, RiOS®, Steelhead®, Think Fast®, Virtual Steelhead®, Whitewater®, Mazu®, Cascade®, Shark®, AirPcap®, BlockStream™, SkipWare®, TurboCap®, WinPcap®, Wireshark®, TrafficScript®, FlyScript™, WWOS™, and Stingray™ are trademarks or registered trademarks of Riverbed Technology, Inc. in the United States and other countries. Riverbed and any Riverbed product or service name or logo used herein are trademarks of Riverbed Technology. All other trademarks used herein belong to their respective owners. The trademarks and logos displayed herein cannot be used without the prior written consent of Riverbed Technology or their respective owners.

Akamai® and the Akamai wave logo are registered trademarks of Akamai Technologies, Inc. SureRoute is a service mark of Akamai. Apple and Mac are registered trademarks of Apple, Incorporated in the United States and in other countries. Cisco is a registered trademark of Cisco Systems, Inc. and its affiliates in the United States and in other countries. EMC, Symmetrix, and SRDF are registered trademarks of EMC Corporation and its affiliates in the United States and in other countries. IBM, iSeries, and AS/400 are registered trademarks of IBM Corporation and its affiliates in the United States and in other countries. Juniper Networks and Junos are registered trademarks of Juniper Networks, Incorporated in the United States and other countries. Linux is a trademark of Linus Torvalds in the United States and in other countries. Microsoft, Windows, Vista, Outlook, and Internet Explorer are trademarks or registered trademarks of Microsoft Corporation in the United States and in other countries. Oracle and JInitiator are trademarks or registered trademarks of Oracle Corporation in the United States and in other countries. UNIX is a registered trademark in the United States and in other countries, exclusively licensed through X/Open Company, Ltd. VMware, ESX, ESXi are trademarks or registered trademarks of VMware, Incorporated in the United States and in other countries.

This product includes Windows Azure Linux Agent developed by the Microsoft Corporation (<http://www.microsoft.com/>). Copyright 2012 Microsoft Corporation.

This product includes software developed by the University of California, Berkeley (and its contributors), EMC, and Comtech AHA Corporation. This product is derived from the RSA Data Security, Inc. MD5 Message-Digest Algorithm.

NetApp Manageability Software Development Kit (NM SDK), including any third-party software available for review with such SDK which can be found at <http://communities.netapp.com/docs/DOC-1152>, and are included in a NOTICES file included within the downloaded files.

For a list of open source software (including libraries) used in the development of this software along with associated copyright and license agreements, see the Riverbed Support site at <https://support.riverbed.com>.

This documentation is furnished "AS IS" and is subject to change without notice and should not be construed as a commitment by Riverbed Technology. This documentation may not be copied, modified or distributed without the express authorization of Riverbed Technology and may be used only in connection with Riverbed products and services. Use, duplication, reproduction, release, modification, disclosure or transfer of this documentation is restricted in accordance with the Federal Acquisition Regulations as applied to civilian agencies and the Defense Federal Acquisition Regulation Supplement as applied to military agencies. This documentation qualifies as "commercial computer software documentation" and any use by the government shall be governed solely by these terms. All other use is prohibited. Riverbed Technology assumes no responsibility or liability for any errors or inaccuracies that may appear in this documentation.



Riverbed Technology
199 Fremont Street
San Francisco, CA 94105

Phone: 415.247.8800
Fax: 415.247.8801
Web: <http://www.riverbed.com>

Part Number
712-00003-19

Contents

Preface.....	1
About This Guide	1
Audience	2
Types of Steelhead Appliances	2
Document Conventions	3
Additional Resources	3
Release Notes	4
Riverbed Documentation and Support Knowledge Base	4
Online Documentation.....	4
Contacting Riverbed.....	4
Internet	4
Technical Support	4
Professional Services	5
Documentation.....	5
What Is New	5
 Chapter 1 - Optimization Techniques and Design Fundamentals.....	7
How Steelhead Appliances Optimize Data	7
Data Streamlining.....	8
Transport Streamlining	9
Application Streamlining	13
Management Streamlining	14
RiOS Data Store Synchronization.....	15
RiOS Data Store Synchronization Requirements	15
RiOS Data Store Error Alarms	15
Choosing the Right Steelhead Appliance.....	16
Deployment Modes for the Steelhead Appliance	17
The Auto-Discovery Protocol.....	18
Original Auto-Discovery Process	19
Configuring Enhanced Auto-Discovery	21
Auto-Discovery and Firewall Considerations	21

Removal of the Riverbed TCP Option Probe.....	21
Stateful Firewall Device in a Multiple In-Path Environment.....	22
Multiple In-Path Discovery Behavior	23
Controlling Optimization	24
In-Path Rules	24
Default In-Path Rules	25
Peering Rules.....	25
The Kickoff and Automatic Kickoff Features	26
Controlling Optimization Configuration Examples	28
Configuring High-Bandwidth, Low-Latency Environment.....	28
Configuring Pass-Through Transit Traffic	30
Fixed-Target In-Path Rules	33
Configuring a Fixed-Target In-Path Rule for an In-Path Deployment	34
Fixed-Target In-Path Rule for an Out-Of-Path Deployment	35
Best Practices for Steelhead Appliance Deployments	36
Chapter 2 - Network Integration Tools.....	39
Redundancy and Clustering	39
Physical In-Path Deployments	39
Virtual In-Path Deployments	40
Out-of-Path Deployments	40
Fail-to-Wire and Fail-to-Block.....	41
Overview of Link State Propagation.....	41
Connection Forwarding	42
Configuring Connection Forwarding.....	43
Multiple-Interface Support Within Connection Forwarding	44
Failure Handling Within Connection Forwarding	44
Connection-Forwarding Neighbor Latency	45
Overview of Simplified Routing.....	45
Chapter 3 - WAN Visibility Modes	49
Overview of WAN Visibility	49
Correct Addressing.....	50
Transparent Addressing.....	51
Port Transparency.....	52
Full Address Transparency	53
Full Address Transparency with Forward Reset.....	55
Implications of Transparent Addressing.....	56
Stateful Systems	56
Network Design Issues	57
Integration into Networks Using NAT.....	60
The Out-of-Band Connection.....	69
Overview of OOB Connections and Addressing Modes.....	70

Configuring OOB Connection Destination Transparency	70
Configuring OOB Connection Full Transparency	71
Configuring WAN Visibility Modes	72
Chapter 4 - QoS Configuration and Integration	75
Overview of Riverbed QoS.....	76
Introduction to Riverbed QoS.....	76
Enforcing QoS Policies using Riverbed QoS	78
Integrating Steelhead Appliances into Existing QoS Architectures	78
WAN-Side Traffic Characteristics and QoS.....	79
QoS Integration Techniques	79
QoS Marking	80
Application Flow Engine.....	83
Overview of Application Flow Engine.....	83
AFE and Microsoft Lync	84
Basic Outbound QoS	84
Configuring Basic Outbound QoS	85
Basic Outbound QoS Mode Restrictions	86
WAN Oversubscription	87
Advanced Outbound QoS	88
QoS Classes.....	88
Choosing a QoS Enforcement System	91
QoS Class Parameters	92
QoS Rules.....	94
Inbound QoS.....	98
Inbound QoS Limitations	99
Inbound QoS Limits	100
Guidelines for the Maximum Number of QoS Classes, Sites, and Rules	100
LAN Bypass.....	103
QoS for IPv6.....	103
QoS in Virtual In-Path and Out-of-Path Deployments	104
QoS in Multiple Steelhead Appliance Deployments.....	104
QoS and Multiple WAN Interfaces	105
QoS Enforcement Best Practices	105
Migrating Between Basic and Advanced Outbound QoS Modes	107
Upgrading to RiOS v6.5 or Later	108
Chapter 5 - QoS Configuration Examples	109
Visualizing and Drawing Your QoS Configuration.....	109
Configuring QoS Using Best Practices.....	113
Example QoS Scenario	113
Configuring QoS on the Data Center Steelhead Appliance	115

Configuring QoS on the Branch Office Steelhead Appliance	119
Configuring QoS Marking on Steelhead Appliances	121
Configuring QoS for Citrix Traffic.....	125
Configuring Basic Outbound QoS and Citrix Traffic in a Pure Interactive Environment	125
Configuring Inbound QoS and Citrix Traffic.....	127
Configuring Advanced Outbound QoS and Citrix Traffic in a Mixed-Traffic Environment ..	131
Configuring QoS and MX-TCP	135
Creating Host Labels	141
Configuring QoS for SSL Common Name Matching	142
Configuring QoS for PCoIP	143
Configuring QoS for SnapMirror	145
Chapter 6 - Path Selection	151
Overview of Path Selection	151
Path Selection Implementation.....	152
Path Selection Properties	152
Example Path Selection Implementation	153
Identifying Traffic Flow Candidates	154
Site Default Path.....	156
Configuring Riverbed Path Selection	157
Valid Path Selection Deployment Design Examples	159
Basic Multiple Route Path Deployment	160
Complex Parallel Path Deployment	162
Complex Single In-Path Interface Deployment	163
Serial Deployment	164
Firewall Path Traversal Deployment	164
Path Selection and Virtual In-Path Deployment	166
Design Validation.....	167
Design Considerations	169
Chapter 7 - Physical In-Path Deployments.....	171
Overview of In-Path Deployment	171
The Logical In-Path Interface	172
In-Path IP Address Selection.....	173
In-Path Default Gateway and Routing	173
Failure Modes.....	174
Fail-to-Wire Mode.....	174
Fail-to-Block Mode	175
Configuring Failure Modes	176
Configuring Link State Propagation	176
Cabling and Duplex.....	177

Choosing the Correct Cables.....	177
Duplex Configuration.....	178
Troubleshooting Cable and Duplex Issues	179
Physical In-Path Deployment Configuration Examples	180
Configuring a Basic Physical In-Path Deployment	180
Configuring a Physical In-Path with Dual Links Deployment.....	182
Configuring a Serial Cluster Deployment with Multiple Links	183
In-Path Redundancy and Clustering Examples	184
Configuring Master and Backup Deployments	184
Configuring Serial Cluster Deployments	187
Configuring Simplified Routing.....	191
Multiple WAN Router Deployments	192
Configuring Multiple WAN Router Deployments Without Connection Forwarding	194
Configuring Multiple WAN Router Deployments with Connection Forwarding.....	198
802.1Q Trunk Deployments.....	206
Overview of VLAN Trunk.....	207
Configuring a Steelhead Appliance on an 802.1Q Trunk Link	208
Capturing Network Traces Using tcpdump	209
Layer-2 WAN Deployments.....	209
Layer-2 WANs.....	209
Broadcast Layer-2 WANs.....	210
VLAN Bridging Deployments	211
Overview of VLAN Bridging Deployment.....	211
VLAN Bridging Considerations.....	212
VLAN Bridging Variations.....	213
Chapter 8 - Virtual In-Path Deployments	217
Overview of Virtual In-Path Deployment.....	217
Configuring an In-Path, Load-Balanced, Layer-4 Switch Deployment	218
Configuring Flow Data Exports in Virtual In-Path Deployments	220
Chapter 9 - WCCP Virtual In-Path Deployments	221
Overview of WCCP	221
WCCP Fundamentals.....	222
Service Groups	222
Assignment Methods	223
Redirection and Return Methods	225
WCCP Clustering and Failover	227
Multiple In-Path WCCP.....	228
The Advantages and Disadvantages of WCCP	228
Configuring WCCP	229
Basic Steps for Configuring WCCP.....	229
Configuring a Simple WCCP Deployment.....	230

Adding a Steelhead Appliance to an Existing WCCP Deployment.....	233
Configuring a WCCP High Availability Deployment.....	234
Configuring a Basic WCCP Router	242
Configuring Additional WCCP Features	243
Specifying the Service Group Password	243
Configuring Multicast Groups	244
Configuring Group Lists to Limit Service Group Members	245
Configuring Access Control Lists.....	246
Configuring Load Balancing in WCCP	249
Flow Data in WCCP	252
Verifying and Troubleshooting WCCP Configurations	252
Chapter 10 - Policy-Based Routing Virtual In-Path Deployments	255
Overview of PBR.....	255
PBR Failover and Cisco Discovery Protocol.....	256
Alternate PBR Failover Mechanisms	257
Connecting the Steelhead Appliance in a PBR Deployment.....	258
Configuring PBR.....	258
Overview of Configuring PBR.....	258
Configuring a Steelhead Appliance to Directly Connect to the Router.....	259
Configuring a Steelhead Appliance to Connect to Layer-2 Switch.....	260
Configuring a Steelhead Appliance to Connect to a Layer-3 Switch.....	262
Configuring a Steelhead Appliance with Object Tracking	263
Configuring a Steelhead Appliance with Multiple PBR Interfaces.....	264
Configuring Multiple Steelhead Appliances to Connect to Multiple Routers	265
Configuring PBR for Load-Balancing WAN Circuits	268
Configuring Local PBR for ICMP Redirection in a Mixed MTU Environment.....	272
Exporting Flow Data and Virtual In-Path Deployments	273
Chapter 11 - IPv6.....	275
Overview of IPv6	275
RiOS RFC Compliance and Feature Compatibility	276
IPv6 Addressing.....	278
Traffic Interception	279
In-Path Rules	280
Deployment Options.....	280
Configuring an In-Path Steelhead Appliance IPv6 Deployment	281
Configuring a Steelhead Appliance Serial Cluster IPv6 Deployment	282
Configuring a Connection Forwarding and Steelhead Appliance IPv6 Deployment.....	284
Configuring a Virtual In-Path Steelhead Appliance IPv6 Deployment	285
Configuring a Fixed-Target Rule Steelhead Appliance IPv6 Deployment	286
Protocol Support	287
Verification and Troubleshooting	287

Chapter 12 - Packet Mode Optimization	289
Overview of Packet Mode Optimization.....	289
Comparison with TCP Proxy Mode Optimization	289
Configuring Packet Mode Optimization.....	290
Design Considerations	294
Best Practices for Packet Mode Optimization.....	295
 Chapter 13 - Satellite Optimization	 297
Overview of Satellite Networks.....	297
Impact of Latency	298
Impact of Loss	298
Satellite Transport Options.....	299
Overview of SCPS.....	299
SCPS Benefits.....	300
Common Uses for SCPS	300
SCPS and Steelhead Appliances.....	300
TCP Optimization for Satellite Environments.....	301
SCPS Discovery	302
Transport Optimization for Satellite Environments	302
Configuring Automatic Detect TCP Optimization.....	305
Integrating the Steelhead Appliance with Existing Satellite Modem TCP Acceleration	306
Licensing SCPS on a Steelhead Appliance.....	306
Configuring Satellite Optimization Features.....	307
Configuring Transport Optimization	307
Configuring Rate Pacing	311
..... Configuring Single-Ended Connection Rule Table Settings	311
Configuring Single-Ended Rules.....	313
Verification and Troubleshooting	315
Analyzing Connection Optimization Information	316
Analyzing Packets for Discovery Probe Stripping	319
Understanding the Health of the Satellite Signal.....	321
Potential Performance Impact of Loss at the Start of Flow	322
Variance in SCPS Performance	322
 Chapter 14 - VPN Routing and Forwarding.....	 323
Overview of NSV with VRF Select.....	323
Virtual Routing and Forwarding	324
NSV with VRF Select	325
IOS Requirements.....	326
Prerequisites for NSV	326
Example NSV Network Deployment	326
Configuring NSV	328
Basic Steps for Configuring NSV	329

Configuring the Data Center Router	329
Configuring the PBR Route Map.....	330
Decoupling VRF from the Subinterface to Implement NSV	331
Configuring Static Routes	331
Configuring the Branch Office Router	332
Configuring the Data Center Steelhead Appliance	333
Configuring the Branch Office Steelhead Appliance	333
Chapter 15 - Out-of-Path Deployments.....	335
Overview of Out-of-Path Deployment	335
Limitations of Out-of-Path Deployments.....	336
Configuring Out-of-Path Deployments.....	337
Chapter 16 - Data Protection Deployments	339
Overview of Data Protection.....	339
Planning for a Data Protection Deployment.....	340
LAN-side Throughput and Data Reduction Requirements	340
Predeployment Questionnaire.....	342
Configuring Steelhead Appliances for Data Protection	345
Adaptive Data Streamlining Feature Settings	346
CPU Settings.....	346
Best Practices for Data Streamlining and Compression.....	347
MX-TCP Settings.....	348
The Steelhead Appliance WAN Buffer Settings	348
The Router WAN Buffer Settings	348
Common Data Protection Deployments	349
Remote Office, Branch Office Backups	349
Network Attached Storage Replication.....	349
Storage Area Network Replication	350
Designing for Scalability and High Availability	351
Overview of N+M Architecture	351
Using MX-TCP in N+M Deployments	351
SnapMirror Optimization	353
Troubleshooting and Fine-Tuning	354
Third-Party Interoperability.....	355
Chapter 17 - Storage Area Network Replication	357
Overview of SAN Replication.....	357
Storage Optimization Modules.....	358
FCIP Optimization Module.....	358
SRDF Optimization Module	361
Best Practices for SAN Replication Using TCP/IP	366

Best Practices for SAN Replication Using FCIP	367
Best Practices for a Cisco MDS FCIP Configuration	367
Best Practices for a Brocade 7800 Configuration.....	371
Best Practices for a Brocade 7500 Configuration.....	377
Chapter 18 - Proxy File Services Deployments	381
Overview of Proxy File Services.....	381
When to Use PFS.....	382
PFS Terms.....	383
Upgrading v2.x PFS Shares	383
Domain and Local Workgroup Settings	384
Domain Mode	384
Local Workgroup Mode.....	385
PFS Share Operating Modes.....	386
Lock Files	387
Configuring PFS.....	387
Chapter 19 - Video Optimization	391
Overview of Video Optimization	391
HTTP Stream Splitting	392
Video On-Demand with HTTP Prepopulation.....	395
Chapter 20 - Authentication, Security, Operations, and Monitoring.....	397
Overview of Authentication.....	397
Authentication Features.....	398
Configuring a RADIUS Server.....	399
Configuring a RADIUS Server with FreeRADIUS	399
Configuring RADIUS Authentication in the Steelhead Appliance.....	400
Configuring RADIUS CHAP Authentication.....	401
Configuring a TACACS+ Server.....	402
Configuring TACACS+ with Cisco Secure Access Control Servers.....	402
Configuring TACACS+ Authentication in the Steelhead Appliance.....	402
Securing Steelhead Appliances.....	403
Overview of Securing Steelhead Appliances	403
Best Practices for Securing Access to Steelhead Appliances	404
Best Practices for Enabling Steelhead Appliance Security Features	410
Best Practices for Policy Controls.....	413
Best Practices for Security Monitoring	413
Configuring SSL Certificates for Web User Interface	414
REST API Access.....	415
Capacity Planning.....	416
Model Characteristics	416

Admission Control	417
Overview of Exporting Flow Data	419
SNMP Monitoring	420
Configuring SNMP v3 Authentication and Privacy	426
Chapter 21 - Troubleshooting Steelhead Appliance Deployment Problems.....	431
Common Deployment Issues.....	431
Duplex Mismatches.....	432
Inability to Access Files During a WAN Disruption	434
Network Asymmetry	434
Unknown (or Unwanted) Steelhead Appliance Appears on the Current Connections List	436
Outdated Antivirus Software	437
Packet Ricochets.....	437
Router CPU Spikes After WCCP Configuration.....	438
Server Message Block Signed Sessions.....	439
Unavailable Opportunistic Locks.....	444
MTU Sizing	445
MTU Issues	446
Determining MTU Size in Deployments.....	446
Connection-Forwarding MTU Considerations	447
Chapter 22 - Steelhead Mobile Deployments	449
Overview of Steelhead Mobile Deployment.....	449
Basic Setup for Deploying Steelhead Mobile.....	449
Steelhead Mobile with VPN Deployments	450
Steelhead Mobile with Firewall Deployments	451
Branch Office and Remote Access Deployments	452
Multiple Mobile Controller Deployments.....	452
Overview of Multiple Mobile Controller Deployments	453
Mobile Controller Concurrent User Limits.....	454
Configuring Multiple Mobile Controllers for Redundancy	454
Preparing to Join Mobile Controllers in a High-Availability Cluster	456
Ports Used with Mobile Controllers and Mobile Clients.....	456
Location Awareness.....	457
Overview of Location Awareness	457
Branch Warming	457
SSL with Steelhead Mobile	459
Traditional SSL Optimization	460
Advanced High-Security SSL Optimization.....	460
Configuring Steelhead Mobile and SSL	461
Multiple Mobile Controllers and SSL	461
Steelhead Mobile Best Practices and Other Considerations.....	462
Deployment Scenarios	462
Management Best Practices.....	463

Licensing Best Practices	464
Antivirus Software	464
Signed SMB Support	465
Optimization Before User Log In	465
Index	467

Preface

Welcome to the *Steelhead Appliance Deployment Guide*. Read this preface for an overview of the information provided in this guide and contact information. This preface includes the following sections:

- [“About This Guide” on page 1](#)
- [“Additional Resources” on page 3](#)
- [“Contacting Riverbed” on page 4](#)
- [“What Is New” on page 5](#)

About This Guide

The *Steelhead Appliance Deployment Guide* describes why and how to configure the Steelhead appliance in complex in-path and out-of-path deployments such as failover, multiple routing points, static clusters, connection forwarding, WCCP, Layer-4, PBR, and PFS. It also includes chapters on the Central Management Console and Steelhead Mobile software.

This guide includes information relevant to the following products:

- Riverbed Optimization System (RiOS)
- Riverbed Steelhead appliance (Steelhead appliance)
- Riverbed Steelhead CX appliance (Steelhead CX)
- Riverbed Steelhead EX appliance (Steelhead EX)
- Riverbed Steelhead DX appliance (Steelhead DX)
- Riverbed Virtual Steelhead appliance (VSH)
- Riverbed Cloud Steelhead (CSH)
- Riverbed Central Management Console (CMC)
- Riverbed Central Management Console Virtual Edition (CMC-VE)
- Riverbed Steelhead Mobile software (Steelhead Mobile)
- Riverbed Steelhead Mobile Controller appliance (Mobile Controller)
- Riverbed Virtual Steelhead Mobile Controller (Virtual SMC)
- Riverbed Steelhead Mobile Client (Mobile Client)

- Riverbed Interceptor appliance (Interceptor appliance)
- Riverbed Virtual Services Platform (VSP)
- Riverbed Services Platform (RSP)
- Riverbed Granite Core appliance (Granite Core)
- Riverbed Whitewater Cloud Storage Appliance (Whitewater appliance)

Audience

This guide is written for storage and network administrators familiar with administering and managing WANs using common network protocols such as TCP, CIFS, HTTP, FTP, and NFS.

You must also be familiar with:

- the Management Console. For details, see the *Steelhead Appliance Management Console User's Guide*.
- connecting to the RiOS CLI. For details, see the *Riverbed Command-Line Interface Reference Manual*.
- the installation and configuration process for the Steelhead appliance. For details, see the *Steelhead Appliance Installation and Configuration Guide* and the *Virtual Steelhead Appliance Installation Guide*.
- the Interceptor appliance. For details, see *Interceptor Appliance User's Guide*.
- the CMC. For details, see *Riverbed Central Management Console User's Guide*.
- the Mobile Controller. For details, see *Steelhead Mobile Controller User's Guide*.

Types of Steelhead Appliances

The Steelhead appliance product line includes several types of devices. Most of the information in the *Steelhead Appliance Deployment Guide* applies to the following appliances:

- **Steelhead appliance (xx50)** - includes WAN optimization and Proxy File service (PFS). RSP is available with an additional license. For details, see the *Steelhead Appliance Management Console User's Guide* for Steelhead appliance (xx50).
- **Steelhead CX (xx55)** - is a WAN optimization-only device. Desktop models have two in-path interfaces. For details, see the *Steelhead Appliance Management Console User's Guide* for Steelhead CX (xx55).
- **Steelhead EX (xx60)** - includes WAN optimization and VSP. The Riverbed Granite product family, which provides branch storage services, is available with an additional license. For details, see the *Steelhead Appliance Management Console User's Guide* for Steelhead EX (xx60).
- **Steelhead DX** - addresses the demands of disaster recovery and data protection over WANs for data center-to-data center, with up to 60-times acceleration of data replication workloads. For details, see the *Steelhead Appliance Management Console User's Guide* for Steelhead DX.
- **VSH** - is a virtualized version of the Steelhead appliance that runs under VMware ESX/ESXi and the Cisco Services-Ready Engine (SRE) platform. For details, see the *Virtual Steelhead Appliance Installation Guide*.
- **CSH** - is the Steelhead appliance for public cloud computing environments. You deploy the CSH differently from the Steelhead appliance and the VSH. For details, see the *Riverbed Cloud Services User's Guide*.

- **Mobile Client** - optimizes network traffic from remote users who are accessing the enterprise network using any type of remote access (dial-up, broadband, wireless, and so on). For details, see the *Steelhead Mobile Controller User's Guide*.
- **Mobile Controller** - provides management functionality for Mobile Clients. For details, see the *Steelhead Mobile Controller User's Guide*.
- **CMC** - provides management functionality for various Riverbed products, including Steelhead appliances, Mobile Controllers, and Interceptor appliances. For details, see the *Riverbed Central Management Console User's Guide*.

For more details on the Steelhead appliance family, see http://www.riverbed.com/us/products/steelhead_appliance/.

Document Conventions

This guide uses the following standard set of typographical conventions.

Convention	Meaning
<i>italics</i>	Within text, new terms, emphasized words, and REST API URIs appear in <i>italic</i> typeface.
boldface	Within text, CLI commands, CLI parameters, and REST API properties appear in bold typeface.
Courier	Code examples appears in Courier font: <pre>amnesiac > enable amnesiac # configure terminal</pre>
< >	Values that you specify appear in angle brackets: interface <ipaddress>
[]	Optional keywords or variables appear in brackets: ntp peer <addr> [version <number>]
{ }	Required keywords or variables appear in braces: {delete <filename>}
	The pipe symbol represents a choice to select one keyword or variable to the left or right of the symbol. The keyword or variable can be either optional or required: {delete <filename> upload <filename>}

Additional Resources

This section describes resources that supplement the information in this guide. It includes the following:

- [“Release Notes” on page 4](#)
- [“Riverbed Documentation and Support Knowledge Base” on page 4](#)
- [“Online Documentation” on page 4](#)

Release Notes

The following online file supplements the information in this guide. It is available on the Riverbed Support site at <https://support.riverbed.com>.

Release Notes	Purpose
<product>_<version_number> <build_number>.pdf	Describes the product release and identifies fixed problems, known problems, and work-arounds. This file also provides documentation information not covered in the manuals or that has been modified since publication.

Examine this file before you begin the installation and configuration process. It includes important information about this release of the Steelhead appliance.

Riverbed Documentation and Support Knowledge Base

For a complete list and the most current version of Riverbed documentation, log in to the Riverbed Support site at <https://support.riverbed.com>.

The Riverbed Knowledge Base is a database of known issues, how-to documents, system requirements, and common error messages. You can browse titles or search for keywords and strings. To access the Riverbed Knowledge Base, log in to the Riverbed Support site at <https://support.riverbed.com>.

Online Documentation

The Riverbed documentation set is periodically updated with new information. To access the most current version of Riverbed documentation and other technical information, consult the Riverbed Support site at <https://support.riverbed.com>.

Contacting Riverbed

This section describes how to contact departments within Riverbed.

Internet

You can learn about Riverbed products at <http://www.riverbed.com>.

Technical Support

If you have problems installing, using, or replacing Riverbed products, contact Riverbed Support or your channel partner who provides support. To contact Riverbed Support, open a trouble ticket by calling 1-888-RVBD-TAC (1-888-782-3822) in the United States and Canada or +1 415 247 7381 outside the United States. You can also go to <https://support.riverbed.com>.

Professional Services

Riverbed has a staff of professionals who can help you with installation, provisioning, network redesign, project management, custom designs, consolidation project design, and custom coded solutions. To contact Riverbed Professional Services, email proserve@riverbed.com or go to <http://www.riverbed.com/services-training/Services-Training.html>.

Documentation

The Riverbed Technical Publications team continually strives to improve the quality and usability of Riverbed documentation. Riverbed appreciates any suggestions you might have about its online documentation or printed materials. Send documentation comments to techpubs@riverbed.com.

What Is New

Since the *Steelhead Appliance Deployment Guide December 2013* release, the following information has been added or updated:

- Updated - [“Configuring Connection Forwarding” on page 43](#)
- Updated - [“Application Flow Engine” on page 83](#)
- New - [“Configuring QoS for SSL Common Name Matching” on page 142](#)
- Updated - [“Path Selection” on page 151](#)
- Updated - [“Multiple WAN Router Deployments” on page 192](#)
- Updated - [“Configuring a WCCP High Availability Deployment” on page 234](#)
- Updated - [“Overview of IPv6” on page 275](#)
- Updated - [“MTU Sizing” on page 445](#)

CHAPTER 1 Optimization Techniques and Design Fundamentals

This chapter describes how the Steelhead appliance optimizes data, the factors you need to consider when designing your Steelhead appliance deployment, and how and when to implement the most commonly used Steelhead appliance features.

This chapter includes the following sections:

- [“How Steelhead Appliances Optimize Data” on page 7](#)
- [“RiOS Data Store Synchronization” on page 15](#)
- [“Choosing the Right Steelhead Appliance” on page 16](#)
- [“Deployment Modes for the Steelhead Appliance” on page 17](#)
- [“The Auto-Discovery Protocol” on page 18](#)
- [“Auto-Discovery and Firewall Considerations” on page 21](#)
- [“Multiple In-Path Discovery Behavior” on page 23](#)
- [“Controlling Optimization” on page 24](#)
- [“Controlling Optimization Configuration Examples” on page 28](#)
- [“Fixed-Target In-Path Rules” on page 33](#)
- [“Best Practices for Steelhead Appliance Deployments” on page 36](#)

How Steelhead Appliances Optimize Data

The Steelhead appliance optimizes data in the following ways:

- [“Data Streamlining” on page 8](#)
- [“Transport Streamlining” on page 9](#)
- [“Application Streamlining” on page 13](#)
- [“Management Streamlining” on page 14](#)

The causes for slow throughput in WANs are well known: high delay (round-trip time or latency), limited bandwidth, and chatty application protocols. Large enterprises spend a significant portion of their information technology budgets on storage and networks, much of it spent to compensate for slow throughput by deploying redundant servers and storage and the required backup equipment. Steelhead appliances enable you to consolidate and centralize key IT resources to save money, simplify key business processes, and improve productivity.

RiOS is the software that powers the Steelhead appliance and Steelhead Mobile. With RiOS, you can solve a range of problems affecting WANs and application performance, including:

- insufficient WAN bandwidth.
- inefficient transport protocols in high-latency environments.
- inefficient application protocols in high-latency environments.

RiOS intercepts client/server connections without interfering with normal client/server interactions, file semantics, or protocols. All client requests are passed through to the server normally, although relevant traffic is optimized to improve performance.

Data Streamlining

With data streamlining, Steelhead appliances and Steelhead Mobile can reduce WAN bandwidth utilization by 65% to 98% for TCP-based applications. This section includes the following topics:

- [“Scalable Data Referencing” on page 8](#)
- [“Bi-Directional Synchronized RiOS Data Store” on page 9](#)
- [“Unified RiOS Data Store” on page 9](#)

Scalable Data Referencing

In addition to traditional techniques like data compression, RiOS also uses a Riverbed proprietary algorithm called Scalable Data Referencing (SDR). RiOS SDR breaks up TCP data streams into *unique data chunks* that are stored on the hard disks (*RiOS data store*) of the device running RiOS (a Steelhead appliance or Steelhead Mobile host system). Each data chunk is assigned a unique integer label (*reference*) before it is sent to a peer RiOS device across the WAN. When the same byte sequence occurs in future transmissions from clients or servers, the reference is sent across the WAN instead of the raw data chunk. The peer RiOS device (a Steelhead appliance or Steelhead Mobile host system) uses this reference to find the original data chunk on its RiOS data store and reconstruct the original TCP data stream.

Files and other data structures can be accelerated by data streamlining even when they are transferred using different applications. For example, a file that is initially transferred through CIFS is accelerated when it is transferred again through FTP.

Applications that encode data in a different format when they transmit over the WAN can also be accelerated by data streamlining. For example, Microsoft Exchange uses the MAPI protocol to encode file attachments prior to sending them to Microsoft Outlook clients. As a part of its MAPI-specific optimized connections, the RiOS decodes the data before applying SDR. This enables the Steelhead appliance to recognize byte sequences in file attachments in their native form when the file is subsequently transferred through FTP or copied to a CIFS file share.

Bi-Directional Synchronized RiOS Data Store

Data and references are maintained in persistent storage in the data store within each RiOS device and are stable across reboots and upgrades. To provide further longevity and safety, local Steelhead appliance pairs optionally keep their data stores fully synchronized bi-directionally at all times. Bi-directional synchronization ensures that the failure of a single Steelhead appliance does not force remote Steelhead appliances to send previously transmitted data chunks. This feature is especially useful when the local Steelhead appliances are deployed in a network cluster, such as a master and backup deployment, a serial cluster, or a WCCP cluster.

For details on master and backup deployments, see [“Redundancy and Clustering” on page 39](#). For details on serial cluster deployments, see [“Configuring Serial Cluster Deployments” on page 187](#). For details on WCCP deployments, see [“WCCP Virtual In-Path Deployments” on page 221](#).

Unified RiOS Data Store

A key Riverbed innovation is the unified data store that data streamlining uses to reduce bandwidth usage. After a data pattern is stored on the disk of a Steelhead appliance or Mobile Controller peer, it can be leveraged for transfers to any other Steelhead appliance or Mobile Controller peer, across all accelerated applications. Data is not duplicated within the RiOS data store, even if it is used in different applications, in different data transfer directions, or with new peers. The unified data store ensures that RiOS uses its disk space as efficiently as possible, even with thousands of remote Steelhead appliances or Mobile Controller peers.

Transport Streamlining

Steelhead appliances use a generic latency optimization technique called transport streamlining. This section includes the following topics:

- [“Overview of Transport Streamlining” on page 9](#)
- [“Connection Pooling” on page 12](#)
- [“TCP Automatic Detection” on page 12](#)
- [“Steelhead Mobile TCP Transport Modes” on page 12](#)
- [“Tuning Steelhead Appliances for High-Latency Links” on page 12](#)
- [“TCP Algorithm Selection” on page 13](#)
- [“WAN Buffers” on page 13](#)

You can find additional information about the transport streaming modes in [“QoS Configuration and Integration” on page 75](#) and [“Satellite Optimization” on page 297](#).

Overview of Transport Streamlining

TCP connections suffer a lack of performance due to delay, loss, and other factors. There are many articles written about and other information available regarding how to choose the client and server TCP settings and TCP algorithms most appropriate for various environments. For example, without proper tuning, a TCP connection might never be able to fill the available bandwidth between two locations. You must consider the TCP window sizes used during the lifespan of a connection. If the TCP window size is not large enough, then the sender cannot consume the available bandwidth. You must also consider packet loss due to congestion or link quality.

In many cases, packet loss is an indication to a TCP congestion avoidance algorithm that there is congestion, and congestion is a signal to the sender to slow down. The sender then can choose at which rate to slow down. The sender can:

- undergo a multiplicative decrease: for example, send at one half the previous rate.
- use other calculations to determine a slightly lower rate, just before the point at which congestion occurred.

A Steelhead appliance deployed on the network can automate much of the manual analysis, research, and tuning necessary to achieve optimal performance, while providing you with options to fine tune. Collectively, these settings in the Steelhead appliance are referred to as transport streamlining. The objective of transport streamlining is to mitigate the effects of WANs between client and server. Transport streamlining uses a set of standards-based and proprietary techniques to optimize TCP traffic between Steelhead appliances. These techniques:

- ensure that efficient retransmission methods are used (such as TCP selective acknowledgments).
- negotiate optimal TCP window sizes to minimize the impact of latency on throughput.
- maximize throughput across a wide range of WAN links.

Additionally, a goal for selecting any TCP setting and congestion avoidance algorithm and using WAN optimization appliances is to find a balance between two extremes: acting fair and cooperative by sharing available bandwidth with co-existing flows on one end of the spectrum, or acting aggressive by trying to achieve maximum throughput at the expense of other flows on the opposite end of the spectrum. Being on the former end indicates that throughput suffers, and being on latter end indicates that your network is susceptible to congestion collapse.

By default, the Steelhead appliances use standard TCP (as defined in RFC 793) to communicate between peers. This type of TCP algorithm is a loss-based algorithm that relies on the TCP algorithm to calculate the effective throughput for any given connection based on packet loss. Alternatively, you can configure Steelhead appliances to use a delay-based algorithm called *bandwidth estimation*. The purpose of bandwidth estimation is to calculate the rate, based on the delay of the link, to recover more gracefully in the presence of packet loss.

In higher-throughput environments you can enable high-speed TCP (HS-TCP), which is a high-speed loss-based algorithm (as defined in RFC 3649) on the Steelhead appliances to achieve high throughput for links with high bandwidth and high latency. This TCP algorithm shifts toward the more aggressive side of the spectrum. Furthermore, you can shift even further toward the aggressive side of the spectrum, sacrificing fairness, by selecting the maximum TCP (MX-TCP) feature for traffic that you want to transmit over the WAN at a rate defined by the QoS class.

Configuring MX-TCP through the QoS settings leverages QoS features to help protect other traffic and gives the you parameters, such as minimum and maximum percentages of the available bandwidth that TCP connections matching the class can consume. Although not appropriate for all environments, MX-TCP can maintain data transfer throughput in which adverse network conditions, such as abnormally high packet loss, impair performance. Data transfer is maintained without inserting error correction packets over the WAN through forward error correction (FEC). MX-TCP effectively handles packet loss without a decrease in throughput typically experienced with TCP.

The TCP algorithms that rely on loss or delay calculations to determine the throughput should have an appropriate-sized buffer. You can configure the buffer size and choose the TCP algorithm on the Transport Settings page. The default buffer is 262,140 bytes, which should cover any connection of 20 Mbps or less with a round-trip delay up to 100ms. This connection speed and round-trip delay comprises most of branch office environments connecting to a data center or hub site.

This is a high-level summary of each Steelhead appliance TCP congestion avoidance algorithm:

- **Standard TCP** - Standard TCP is a standards-based implementation of TCP and is the default setting in the Steelhead appliance. Standard TCP is a WAN-friendly TCP stack and is not aggressive towards other traffic. Additionally, standard TCP benefits from the higher TCP WAN buffers, which are used by default for each connection between Steelhead appliances.
- **Bandwidth estimation** - Bandwidth estimation is the delay-based algorithm that incorporates many of the features of standard TCP and includes calculation of RTT and bytes acknowledged. This additional calculation avoids the multiplicative decrease in rate detected in other TCP algorithms in the presence of packet loss. Bandwidth estimation is also appropriate for environments in which there is variable bandwidth and delay.
- **HighSpeed TCP (HS-TCP)** - HS-TCP is efficient in long fat networks (LFNs) in which you have large WAN circuits (50 Mbps and above) over long distances. Typically, you use HS-TCP when you have a few long-lived replicated or backup flows. HS-TCP is designed for high-bandwidth and high-delay networks that have a low rate of packet loss due to corruption (bit errors). HS-TCP has a few advantages over standard TCP for LFNs. Standard TCP will *backoff* (slow down the transmission rate) in the presence of packet loss, causing connections to under use the bandwidth.

Also, standard TCP is not as aggressive during the TCP slow-start period to rapidly grow to the available bandwidth. HS-TCP uses a combination of calculations to rapidly fill the link and minimize backoff in the presence of loss. These techniques are documented in RFC 3649. HS-TCP is not beneficial for satellite links because the TCP congestion window recovery requires too many round trips or is too slow. HS-TCP requires that you adjust WAN buffers on the Steelhead appliances to be equal to $2 \times \text{BDP}$, where bandwidth-delay product (BDP) is the product of the WAN bandwidth and round-trip latency between locations. For more specific settings, see [“Storage Area Network Replication” on page 357](#).

- **SkipWare Space Communications Protocol Standards (SCPS) per-connection** - SCPS per connection is for satellite links with little or no packet drops due to corruption. SCPS per connection requires a separate license.

For more details, see [“SCPS Per Connection” on page 304](#).

- **SCPS error tolerance** - SCPS error tolerance is for satellite links that have packet drops due to corruption. You must have a separate license to activate SCPS error tolerance.

For more details, see [“SCPS Error Tolerance” on page 304](#).

This is a high-level summary of additional modes that alter the Steelhead appliance TCP congestion avoidance algorithm:

- **MX-TCP** - MX-TCP is ideal for dedicated links, or to compensate for poor link quality (propagation issues, noise, and so on) or packet drops due to network congestion. The objective of MX-TCP is to achieve maximum TCP throughput. MX-TCP alters TCP by disabling the congestion control algorithm and sending traffic up to a rate you configure, regardless of link conditions. Additionally, MX-TCP can share any excess bandwidth with other QoS classes through adaptive MX-TCP. MX-TCP requires knowledge of the amount of bandwidth available for a given QoS class because, provided that enough traffic matches the QoS class, connections using MX-TCP attempt to consume the bandwidth allotted without regard to any other traffic.

For more details, see [“MX-TCP” on page 93](#) and [“MX-TCP Settings” on page 348](#).

- **Rate pacing** - Rate pacing is a combination of MX-TCP and a TCP congestion avoidance algorithm. You use rate pacing commonly in satellite environments, but you can use it in terrestrial connections as well. The combination of MX-TCP and a TCP congestion avoidance algorithm allows rate pacing to take the best from both features. Rate pacing leverages the rate configured for an MX-TCP QoS class to minimize buffer delays, but can adjust to the presence of loss due to network congestion.

For more details, see [“Configuring Rate Pacing” on page 311](#).

For additional information about transport streamlining mode options, see the following:

- [“Connection Pooling” on page 12](#)
- [“TCP Automatic Detection” on page 12](#)

Connection Pooling

Connection pooling adds a benefit to transport streamlining by minimizing the time for an optimized connection to setup.

Some application protocols, such as HTTP, use many rapidly created, short-lived TCP connections. To optimize these protocols, Steelhead appliances create pools of idle TCP connections. When a client tries to create a new connection to a previously visited server, the Steelhead appliance uses a TCP connection from its pool of connections. Thus the client and the Steelhead appliance do not have to wait for a three-way TCP handshake to finish across the WAN. This feature is called *connection pooling*. Connection pooling is available only for connections using the correct addressing WAN visibility mode.

Transport streamlining ensures that there is always a one-to-one ratio for active TCP connections between Steelhead appliances and the TCP connections to clients and servers. Regardless of the WAN visibility mode in use, Steelhead appliances do not tunnel or perform multiplexing and demultiplexing of data across connections.

For details on correct addressing modes, see [“WAN Visibility Modes” on page 49](#). For details on HTTP optimization, see the *Steelhead Appliance Deployment Guide - Protocols*.

TCP Automatic Detection

One best practice you can consider for nearly every deployment is the TCP automatic detection feature on the data center Steelhead appliances. This feature allows the data center Steelhead appliance to reflect the TCP algorithm in use by the peer. The benefit is that you can select the appropriate TCP algorithm for the remote branch office, and the data center Steelhead appliance uses that TCP algorithm for connections. If Steelhead appliances on both sides of an optimized connection use the automatic detection feature, then standard TCP is used.

Steelhead Mobile TCP Transport Modes

This section briefly describes specific transport streamlining modes that operate with Steelhead Mobile.

HS-TCP is not the best choice for interoperating in a Steelhead Mobile environment because it is designed for LFNs (high bandwidth and high delay). Essentially, the throughput is about equal to standard TCP.

MX-TCP is a sender-side modification (configured on the server side) and is used to send data at a specified rate. When Steelhead Mobile is functioning on the receiving side, it can be difficult to deploy MX-TCP on the server side. The issue is defining a sending rate in which it might not be practical to determine the bandwidth that a client can receive on their mobile device because it is unknown and variable.

Tuning Steelhead Appliances for High-Latency Links

Riverbed recommends that you gather WAN delay (commonly expressed as RTT), packet-loss rates, and link bandwidth to better understand the WAN characteristics so that you can make adjustments to the default transport streamlining settings. Also, understanding the types of workloads (long-lived, high-throughput, client-to-server traffic, mobile, and so on) is valuable information for you to appropriately select the best transport streamlining settings.

Specific settings for high-speed data replication are covered in [“Storage Area Network Replication” on page 357](#). The settings described in this chapter approximate when you can adjust the transport streamlining settings to improve throughput.

TCP Algorithm Selection

The default Steelhead appliance settings are appropriate in most deployment environments. Based on RTT, bandwidth, and loss, you can optionally choose different transport streamlining settings. A solid approach to selecting the TCP algorithm found on the Transport Settings page is to use the automatic detection feature (auto-detect) on the data center Steelhead appliance. The benefit to automatic detection is that the data center Steelhead appliance reflects the choice of TCP algorithm in use at the remote site. You select the TCP algorithm at the remote site based on WAN bandwidth, RTT, and loss.

A general guideline is that any connection over 50 Mbps can benefit from using HS-TCP, unless connection is over satellite (delay greater than 500 ms). You can use MX-TCP for high data rates if the end-to-end bandwidth is known and dedicated.

When you are factoring in loss at lower-speed circuits, consider using bandwidth estimation. When planning, consider when packet loss is greater than 0.1%. Typically, MPLS networks are below 0.1% packet loss, while other communication networks can be higher. For any satellite connection, the appropriate choices are SCPS (if licensed) or bandwidth estimation.

For specific implementation details, see [“Satellite Optimization” on page 297](#).

WAN Buffers

After you select the TCP algorithm, another setting to consider is the WAN-send and WAN-receive buffers. You can use bandwidth and RTT to determine the BDP. BDP is a multiplication of bandwidth and RTT, and is commonly divided by 8 and expressed in bytes. To get better performance, the Steelhead appliance as a TCP proxy typically uses two times BDP as its WAN-send and WAN-receive buffer. For asymmetry, you can have the WAN-send buffer reflect the bandwidth and delay in the transmit direction, while the WAN-receive buffer reflects the bandwidth and delay in the receive direction. Note that you do not have to adjust the buffer settings unless there is a relatively small number of connections and you want to consume most or all of the available WAN bandwidth.

Application Streamlining

You can apply application-specific optimization for specific application protocols. For Steelhead appliances using RiOS v6.0 or later, application streamlining includes:

- CIFS for Windows and Mac clients (Windows file sharing, backup and replication, and other Windows-based applications).
- MAPI, including encrypted email (Microsoft Exchange v 5.5, 2000, 2003, and 2007).
- NFS v3 for UNIX file sharing.
- TDS for Microsoft SQL Server.
- HTTP.
- HTTPS and SSL.
- IMAP-over-SSL.
- Oracle 9i, which comes with Oracle Applications 11i.
- Oracle10gR2, which comes with Oracle E-Business Suite R12.
- Lotus Notes v6.0 or later.
- Encrypted Lotus Notes.

- ICA Client Drive Mapping.
- support for multistream and multiport ICA.

Protocol-specific optimization reduces the number of round trips over the WAN for common actions and help move through data obfuscation and encryption by:

- opening and editing documents on remote file servers (CIFS).
- sending and receiving attachments (MAPI and Lotus Notes).
- viewing remote intranet sites (HTTP).
- securely performing RiOS SDR for SSL-encrypted transmissions (HTTPS).

For more details on application streamlining, see the *Steelhead Appliance Deployment Guide - Protocols*.

Management Streamlining

Developed by Riverbed, management streamlining simplifies the deployment and management of RiOS devices. This includes both hardware and software:

- **Auto-Discovery Protocol** - Auto-discovery enables Steelhead appliances and Steelhead Mobile to automatically find remote Steelhead appliances, and begin to optimize traffic. Auto-discovery avoids the requirement of having to define lengthy and complex network configurations on Steelhead appliances. The auto-discovery process enables administrators to:
 - control and secure connections.
 - specify which traffic is to be optimized.
 - specify peers for optimization.

For more information, see [“The Auto-Discovery Protocol” on page 18](#).

- **CMC** - The CMC enables new, remote Steelhead appliances to be automatically configured and monitored. It also provides a single view of the overall optimization benefit and health of the Steelhead appliance network.
- **Mobile Controller** - The Mobile Controller is the management appliance that you use to track the individual health and performance of each deployed software client and to manage enterprise client licensing. The Mobile Controller enables you to see who is connected, view their data reduction statistics, and perform support operations such as resetting connections, pulling logs, and automatically generating traces for troubleshooting. You can perform all of these management tasks without end-user input.

For more information, see [“Steelhead Mobile Deployments” on page 449](#).

RiOS Data Store Synchronization

RiOS data store synchronization enables pairs of local Steelhead appliances to synchronize their data stores with each other, even while they are optimizing connections. RiOS data store synchronization is typically used to ensure that if a Steelhead appliance fails, no loss of potential bandwidth savings occurs, because the data segments and references are on the other Steelhead appliance. This section includes the following topics:

- [“RiOS Data Store Synchronization Requirements” on page 15](#)
- [“RiOS Data Store Error Alarms” on page 15](#)

You can use RiOS data store synchronization for physical in-path, virtual in-path, or out-of-path deployments. You enable synchronization on two Steelhead appliances, one as the synchronization master, and the other as the synchronization backup.

The traffic for RiOS data store synchronization is transferred through either the Steelhead appliance primary or auxiliary network interfaces, not the in-path interfaces.

RiOS data store synchronization is a bidirectional operation between two Steelhead appliances, regardless of which deployment model you use. The Steelhead appliance *master* and *backup* designation is relevant only in the initial configuration, when the master Steelhead appliance RiOS data store essentially overwrites the backup Steelhead appliance RiOS data store.

RiOS Data Store Synchronization Requirements

The synchronization master and its backup:

- must have the same hardware model.
- must be running the same version of RiOS.
- do not have to be in the same physical location. If they are in different physical locations, they must be connected via a fast, reliable LAN connection with minimal latency.

Important: Before you replace a synchronization master for any reason, Riverbed recommends that you make the synchronization backup the new synchronization master. This is so that the new master (the former backup) can warm the new (replacement) Steelhead appliance, ensuring that the most data is optimized and none is lost.

RiOS Data Store Error Alarms

You receive an email notification if an error occurs in the RiOS data store. The RiOS data store alarms are enabled by default.

If the alarm was caused by an unintended change to the configuration, you can change the configuration to match the old RiOS data store settings again and then restart the optimization service (without clearing the data store) to reset the alarm. In certain situations you might need to clear the RiOS data store. Typical configuration changes that require a clear data store are changes to the data store encryption type or enabling the extended peer table.

To clear the data store of data restart the optimization service and click Clear the Data Store.

For more details on the RiOS data store error alarm, see the *Steelhead Appliance Management Console User's Guide*.

Choosing the Right Steelhead Appliance

Generally, you select a Steelhead appliance model based on the number of users, the bandwidth requirements, and the applications used at the deployment site. However:

- if you do not want to optimize applications that transfer large amounts of data (for example, WAN-based backup or restore operations, system image, or update distribution), choose your Steelhead appliance model based on the amount of bandwidth and number of connections at your site.
- if you do want to optimize applications that transfer large amounts of data, choose your Steelhead appliance model based on the amount of bandwidth and number of connections at your site, and on the size of the RiOS data store.

After you consider these factors, you might also consider high availability, redundancy, data protection, or other requirements.

If no single Steelhead appliance model meets your requirements, depending on your deployment model, there are many ways to cluster Steelhead appliances together to provide scaling, and if necessary, redundancy. Steelhead appliance models vary according to the following attributes:

- Number of concurrent TCP connections that can be optimized
- Amount of disk storage available for RiOS SDR
- Amount of WAN bandwidth that can be used for optimized bandwidth
- Maximum possible in-path interfaces
- Availability of fiber interfaces
- Availability of RAID for RiOS data store
- Availability of redundant power supplies
- Upgrade options through software licenses
- Support for PFS shares
- Possibility of 64-bit RSP images

All Steelhead appliance models have the following specifications that are used to determine the amount of traffic that a single Steelhead appliance can optimize:

- **Number of concurrent TCP connections** - Each Steelhead appliance model can optimize a certain number of concurrent TCP connections.

The number of TCP connections you need for optimization depends on the number of users at your site, the applications that you use, and whether you want to optimize all applications or just a few of them. When planning corporate enterprise deployments, Riverbed recommends that you use ratios of 5-15 connections per user if full optimization is desired, depending on the applications being used.

If the number of connections that you want to optimize exceeds the limit of the Steelhead appliance model, the Steelhead appliance allows excess connections to pass through unoptimized.

The TCP protocol only supports approximately 64,000 ports per IP address for outbound or inbound connections to or from a unique IP-Port pair. If your Steelhead appliance supports a higher than 64,000 connections, Riverbed recommends that you use multiple in-path interfaces or service port mapping.

For more details on concurrent TCP connections, see https://supportkb.riverbed.com/support/index?page=content&id=S:S16309&actp=search&viewlocale=en_US&searchid=1342759374914.

- **WAN bandwidth rating** - Each Steelhead appliance model has a limit on the rate at which it pushes optimized data toward the WAN. You might not need a Steelhead appliance model that is rated for the same bandwidth available at the deployment site, but Riverbed recommends that you make sure that the selected appliance is not a bottleneck for the outbound optimized traffic. This limit does not apply to pass-through traffic.

When a Steelhead appliance reaches its rate limit, it does not start passing through traffic, but it begins shaping optimized traffic to this limit. New optimized connections can be set up if the connection limit allows.

- **RiOS data store size** - Each Steelhead appliance model has a fixed amount of disk space available for RiOS SDR. Because SDR stores unique patterns of data, the amount of data store space needed by a deployed Steelhead appliance differs from the amount needed by applications or file servers. For the best optimization possible, the RiOS data store must be large enough to hold all of the commonly accessed data at a site. Old data that is recorded in the RiOS data store might eventually be overwritten by new data, depending on traffic patterns.

At sites where applications transfer large amounts of data (for example, WAN-based backup or restore operations, system image, or update distribution), you must not select the Steelhead appliance model based only on the amount of bandwidth and number of connections at the site, but also on the size of RiOS data store. Sites without these applications are typically sized by considering the bandwidth and number of connections.

If you need help planning, designing, deploying, or operating your Steelhead appliances, Riverbed offers consulting services directly and through Riverbed authorized partners. For details contact, Riverbed Professional Services by email at email proserve@riverbed.com or go to <http://www.riverbed.com/services-training/Services-Training.html>.

Deployment Modes for the Steelhead Appliance

You can deploy Steelhead appliances into the network in many different ways. Deployment modes available for the Steelhead appliances include:

- **Physical In-Path** - In a physical in-path deployment, the Steelhead appliance is physically in the direct path between clients and servers. In-path designs are the simplest to configure and manage, and they are the most common type of Steelhead appliance deployment, even for large sites. Many variations of physical in-path deployments are possible to account for redundancy, clustering, and asymmetric traffic flows.

For details, see [“Physical In-Path Deployments” on page 171](#).

- **Virtual In-Path** - In a virtual in-path deployment, you can use a redirection mechanism (like WCCP, PBR, or Layer-4 switching) to place the Steelhead appliance virtually in the path between clients and servers.

For details, see [“Virtual In-Path Deployments” on page 217](#).

- **Out-of-Path** - In an out-of-path deployment, the Steelhead appliance is not in the direct path between the client and the server. In an out-of-path deployment, the Steelhead appliance acts as a proxy. This type of deployment might be suitable for locations where physical in-path or virtual in-path configurations are not possible. However, out-of-path deployments have several drawbacks that you must be aware of.

For details, see [“Out-of-Path Deployments” on page 335](#).

The Auto-Discovery Protocol

This section describes the Steelhead appliance auto-discovery protocol. This section includes the following topics:

- [“Original Auto-Discovery Process” on page 19](#)
- [“Configuring Enhanced Auto-Discovery” on page 21](#)

Auto-discovery enables Steelhead appliances to automatically find remote Steelhead appliances and to optimize traffic with them. Auto-discovery relieves you of having to manually configure the Steelhead appliances with large amounts of network information.

The auto-discovery process enables you to:

- control and secure connections.
- specify which traffic is optimized.
- specify how remote peers are selected for optimization.

The types of auto-discovery are as follows:

- **Original Auto-Discovery** - Automatically finds the first remote Steelhead appliance along the connection path.
- **Enhanced Auto-Discovery** (available in RiOS v4.0.x or later) - Automatically finds the last Steelhead appliance along the connection path.

Most Steelhead appliance deployments use auto-discovery. You can also manually configure Steelhead appliance pairing using fixed-target in-path rules, but this approach requires ongoing configuration. Fixed-target rules also require tracking new subnets that are present in the network and for which Steelhead appliances are responsible for optimizing the traffic.

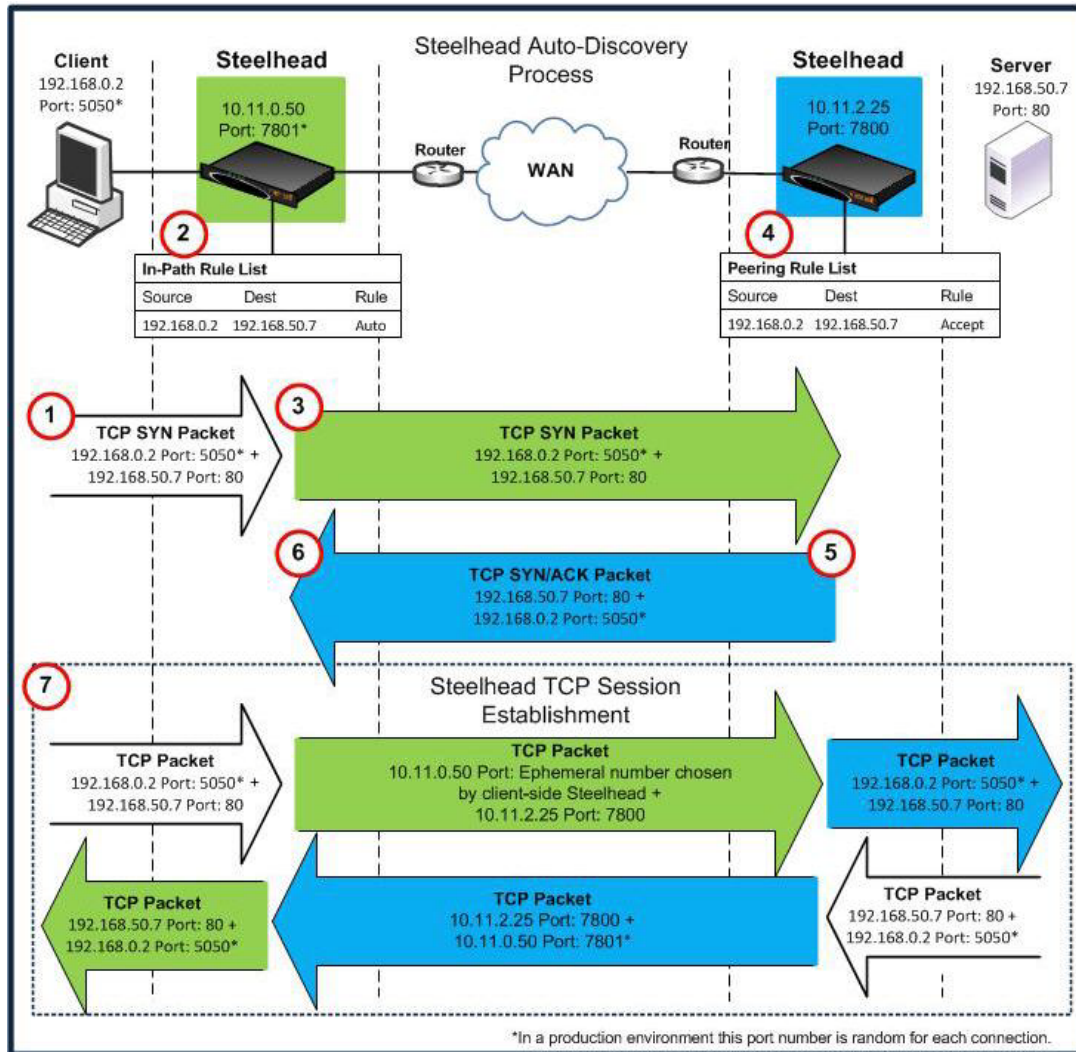
You can use auto-discovery if the Steelhead appliances are deployed physically in-path or virtually in-path (such as WCCP or PBR). The following describes physically in-path Steelhead appliances, but the packet flow is identical with a virtual in-path deployment.

For details on fixed-target in-path rules, see [“Fixed-Target In-Path Rules” on page 33](#).

Original Auto-Discovery Process

The following section describes how a client connects to a remote server when the Steelhead appliances have auto-discovery enabled. In this example, each Steelhead appliance uses correct addressing and a single subnet.

Figure 1-1. The Auto-Discovery Process



Note: This example does not show asymmetric routing detection or enhanced auto-discovery peering.

In the original auto-discovery process:

1. The client initiates the TCP connection by sending a TCP SYN packet.

2. The client-side Steelhead appliance receives the packet on its LAN interface, examines the packet, discovers that it is a SYN, and continues processing the packet:
 - Using information from the SYN packet (for example, the source or destination address, or VLAN tag), the Steelhead appliance performs an action based on a configured set of rules, called *in-path rules*. In this example, because the matching rule for the packet is set to auto, the Steelhead appliance uses auto-discovery to find the remote Steelhead appliance.
 - The Steelhead appliance appends a TCP option to the packet TCP option field. This is the probe query option. The probe query option contains the in-path IP address of the client-side Steelhead appliance. Nothing else in the packet changes, only the option is added.
3. The Steelhead appliance forwards the modified packet (denoted as SYN_probe_query) out of the WAN interface. Because neither the source or destination fields are modified, the packet is routed in the same manner as if there was no Steelhead appliance deployed.
4. The server-side Steelhead appliance receives the SYN_probe_query packet on its WAN interface, examines the packet, discovers that it is a SYN packet, and therefore searches for a TCP probe query. If found, the server-side Steelhead appliance:
 - uses the packet fields and the IP address of the client-side Steelhead appliance to determine what action to take, based on its peering rules. In this example, because the matching rule is set to accept (or auto, depending on the RiOS version), the server-side Steelhead appliance communicates to the client-side Steelhead appliance that it is the remote optimization peer for this TCP connection.
 - removes the probe_query option from the packet, and replaces it with a probe_response option (the probe_query and probe_response use the same TCP option number). The probe_response option contains the in-path IP address of the server-side Steelhead appliance.
 - reverses all of the source and destination fields (TCP and IP) in the packet header. The packet sequence numbers and flags are modified to make the packet look like a normal SYN/ACK server response packet.

If no server-side Steelhead appliances are present, the server ignores the TCP probe that was added by the client-side Steelhead appliance, responds with a regular SYN/ACK resulting in a pass-through connection, and sends the SYN/ACK.
5. The server-side Steelhead appliance transmits the packet to the client-side Steelhead appliance. Because the destination IP address of the packet is now the client IP address, the packet is routed through the WAN just as if the server was responding to the client.
6. The client-side Steelhead appliance receives the packet on its WAN interface, examines it, and discovers that it is a SYN/ACK. The client-side Steelhead appliance scans for and finds the probe_response field, and reads the in-path IP address of the server-side Steelhead appliance. Now the client-side Steelhead appliance knows all the parameters of the packet TCP flow, including the:
 - IP addresses of the client and server.
 - TCP source and destination ports for this connection.
 - in-path IP address of the server-side Steelhead appliance for this connection.
7. The Steelhead appliances now establish three TCP connections:
 - The client-side Steelhead appliance completes the TCP connection setup with the client, as if it were the server.
 - The Steelhead appliances complete the TCP connection between each other.

- The server-side Steelhead appliance completes the TCP connection with the server, as if it were the client.

After the three TCP connections are established, optimization begins. The data sent between the client and server for this specific connection is optimized and carried on its own individual TCP connection between the Steelhead appliances.

Configuring Enhanced Auto-Discovery

In RiOS v4.0.x or later, enhanced auto-discovery is available. Enhanced auto-discovery automatically discovers the last Steelhead appliance in the network path of the TCP connection. In contrast, the original auto-discovery protocol automatically discovers the first Steelhead appliance in the path. The difference is only seen in environments where there are three or more Steelhead appliances in the network path for connections to be optimized.

Enhanced auto-discovery works with Steelhead appliances running the original auto-discovery protocol. Enhanced auto-discovery ensures that a Steelhead appliance optimizes only TCP connections that are being initiated or terminated at its local site, and that a Steelhead appliance does not optimize traffic that is transiting through its site.

For details on passing through transit traffic using enhanced auto-discovery and peering rules, see [“Configuring Pass-Through Transit Traffic” on page 30](#).

To enable enhanced auto-discovery

- Connect to the Steelhead CLI and enter the following commands:

```
enable
configure terminal
in-path peering auto
```

Auto-Discovery and Firewall Considerations

This section contains factors to consider when using auto-discovery:

- [“Removal of the Riverbed TCP Option Probe” on page 21](#)
- [“Stateful Firewall Device in a Multiple In-Path Environment” on page 22](#)

Removal of the Riverbed TCP Option Probe

The most common reason that auto-discovery fails is because a device (typically security related) strips out the TCP options from optimized packets. Auto-discovery relies on using TCP options to determine if a remote Steelhead appliance exists. Common devices that remove TCP options are firewalls and satellite routers.

The Riverbed Support site has knowledge base articles that show example configurations to allow Riverbed auto-discovery options through different firewall types.

To solve the problem, you can configure the devices to ignore or prevent stripping out the TCP option. Alternatively, you can use fixed-target rules on the Steelhead appliances to bypass the auto-discovery process.

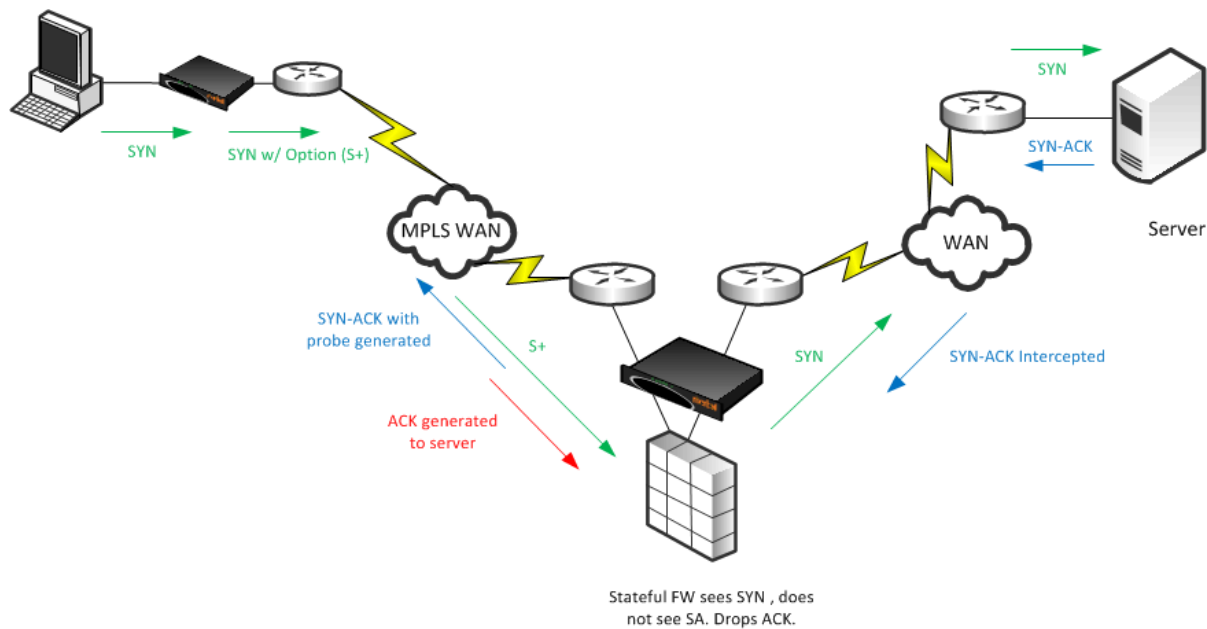
For details, see [“Configuring a Fixed-Target In-Path Rule for an In-Path Deployment” on page 34](#).

Stateful Firewall Device in a Multiple In-Path Environment

Consider the following when you install a Steelhead appliance in which a stateful firewall exists in a multiple in-path environment. A stateful firewall on the LAN-side of a Steelhead appliance might not detect the entire TCP handshake conversation, and this causes the firewall to drop packets.

Figure 1-2 shows an example when the Steelhead appliance is at a transit site where traffic passes through multiple interfaces on the Steelhead appliance, with a firewall located on the LAN. The firewall does not see the whole TCP handshake.

Figure 1-2. Multiple In-Path Interfaces with Stateful Firewall on the LAN



In this example, the stateful firewall detects the SYN packet to the server, but the SYN-ACK response from the server is intercepted at the Steelhead appliance prior to reaching the firewall. When the server-side Steelhead appliance originates the ACK to the server, the stateful firewall often denies the connection because it has not detected the proper SYN-ACK response.

Note: Although the Steelhead appliance can appear to be bridging traffic between its in-path interfaces, this is not true. The Steelhead appliance always generates packets from the owner interface, but it intercepts packets on any in-path interface. This is the necessary behavior in asymmetric routing environments.

When you have a stateful firewall device in a multiple in-path environment, the goal of the Steelhead appliance is to provide optimization for local traffic only at the transit site. The remote site usually has its own local Steelhead appliance if optimization is required. To prevent the transit Steelhead appliance from participating in the auto-discovery process, you can use peering rules to accept only probed SYNs destined to devices at the transit site. Contact Riverbed Professional Services for further options and solutions.

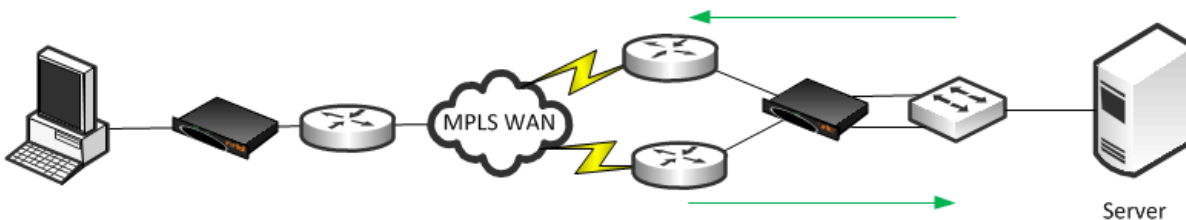
For more details on peering rules, see [“Peering Rules” on page 25](#).

Multiple In-Path Discovery Behavior

This section describes how multiple in-path interfaces on a Steelhead appliance interact to intercept and originate packets for optimized connections. The Steelhead appliance can support up to ten in-path interfaces, depending on the model. The packet flow is the same no matter how many interfaces are used.

You can use multiple in-path interfaces on a Steelhead appliance to allow different network paths in an asymmetric routing environment. For example, you can have two or more routers that have circuits to the same or different MPLS circuits, and you can route traffic over either path. [Figure 1-3](#) shows multiple in-path interfaces.

Figure 1-3. Multiple In-Path Interface Example



[Figure 1-3](#) shows traffic flowing to the server through a different router than traffic returning from the server. In many cases, you can use the two circuits for load balancing, so inbound or outbound traffic can legitimately flow through either router. In [Figure 1-3](#), the Steelhead appliance on the right uses two in-path interfaces to allow for the possibility of asymmetric traffic.

The Steelhead appliance intercepts packets for optimized connections on any in-path interface. The most common reason for enabling multiple in-path interfaces is to ensure that asymmetric routing does not prevent the Steelhead appliance from detecting all packets for an optimized connection.

Packet origination from the Steelhead appliance always comes from one in-path interface. Each of the optimizing Steelhead appliances has one in-path interface, which is referred to as the *owner interface*. The owner interface is bound to the optimized connection. On the client-side, the first Steelhead appliance's last in-path interface to detect the SYN packet from the client is the owner interface. The client-side Steelhead appliance appends a TCP option to this SYN packet, known as a Riverbed probe.

If the SYN packet with a probe passes through the client-side Steelhead appliance, the client-side Steelhead appliance updates the IP address inside the probe option with the IP address of the in-path interface it is passing through. With original auto-discovery, the owner interface is the first in-path interface of the first server-side Steelhead appliance. With enhanced auto-discovery, because the goal is to optimize to the last Steelhead appliance in the path, the owner interface on the server-side is the last Steelhead appliance's first in-path interface to see the probed SYN. In most cases, the actual owner interface does not matter as much as which Steelhead appliances are performing the optimization.

Figure 1-4 shows a more detailed example of in-path interfaces. The server-side Steelhead appliance is on the right.

Figure 1-4. Multiple In-Path Interface (More Detailed Example)

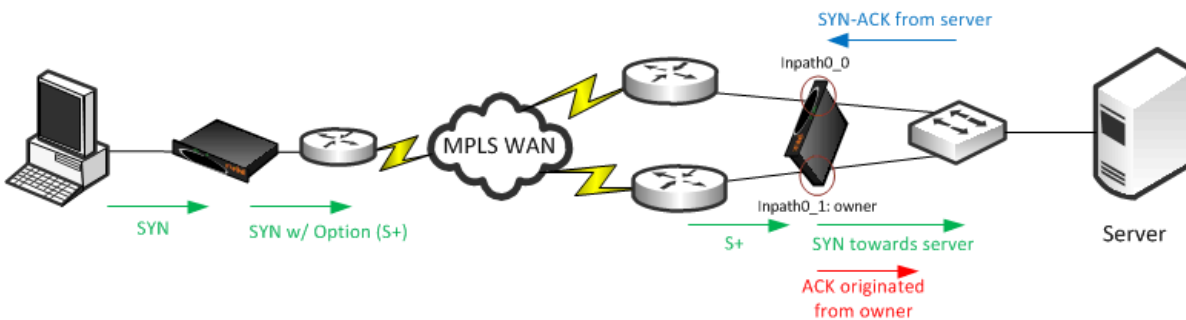


Figure 1-4 shows the client-side Steelhead appliance owner interface is inpath0_0, because this is the interface that detects the client SYN packet. The probed SYN packet reaches the server-side Steelhead appliance on inpath0_1, and therefore inpath0_1 on the server-side Steelhead appliance is the owner. The Steelhead appliance intercepts packets for the optimized connections it is aware of. Even though the server SYN-ACK response comes back into inpath0_0, the Steelhead appliance intercepts these packets, and prevents asymmetric routing from bypassing the Steelhead appliance. Packets originating from the Steelhead appliance are always sourced from the owner interface. The ACK towards the server originates from the server-side Steelhead appliance's inpath0_1 interface.

Controlling Optimization

You can configure what traffic a Steelhead appliance optimizes and what other actions it performs, using the following:

- **In-Path rules** - In-path rules determine the action a Steelhead appliance takes when a connection is initiated, usually by a client.
- **Peering rules** - Peering rules determine how a Steelhead appliance reacts when it detects a probe query.

This section includes the following topics:

- [“In-Path Rules” on page 24](#)
- [“Default In-Path Rules” on page 25](#)
- [“Peering Rules” on page 25](#)
- [“The Kickoff and Automatic Kickoff Features” on page 26](#)

In-Path Rules

In-path rules are used only when a connection is *initiated*. Because connections are typically initiated by clients, in-path rules are configured for the initiating, or client-side, Steelhead appliance. In-path rules determine Steelhead appliance behavior with SYN packets.

In-path rules are an ordered list of fields that a Steelhead appliance attempts to match with SYN packet fields (for example, source or destination subnet, IP address, VLAN, or TCP port). Each in-path rule has an *action* field. When a Steelhead appliance finds a matching in-path rule for a SYN packet, the Steelhead appliance treats the packet according to the action specified in the in-path rule.

The in-path rule actions, each with different configuration possibilities, are as follows:

- **Auto** - Uses the auto-discovery process to determine if a remote Steelhead appliance is able to optimize the connection that this SYN packet.
- **Pass** - Allows the SYN packet to pass through the Steelhead appliance. No optimization is performed on the TCP connection initiated by this SYN packet is trying to initiate.
- **Fixed-Target** - Omits the auto-discovery process and instead uses a specified remote Steelhead appliance as an optimization peer. Fixed-target rules require the input of at least one remote target Steelhead appliance; you can specify an optional backup Steelhead appliance.

For details on fixed-target in-path rules, see [“Fixed-Target In-Path Rules” on page 33](#).

- **Deny** - Drops the SYN packet and sends a message back to its source.
- **Discard** - Drops the SYN packet silently.

Only use in-path rules in the following scenarios:

- TCP SYN packet arrives on the LAN interface of physical in-path deployments.
- TCP SYN packet arrives on the WAN interface of virtual in-path deployments.

Again, both of these scenarios are associated with the first, or *initiating*, SYN packet of the connection. In-path rules are applicable to only the client-side Steelhead appliance. In-path rules have no effect on connections that are already established, regardless of whether the connections are being optimized.

In-path rule configurations differ depending on the action. For example, both the fixed-target and the auto-discovery actions allow you to choose configurations such as what type of optimization is applied, what type of data reduction is used, and what type of latency optimization is applied.

For an example of how in-path rules are used, see [“Configuring High-Bandwidth, Low-Latency Environment” on page 28](#).

Default In-Path Rules

The Steelhead appliance ships with three default in-path rules. Default rules pass through certain types of traffic unoptimized because these protocols (telnet, SSH, HTTPS) are typically used when you deploy and configure your Steelhead appliances. The default in-path rules can be removed or overwritten by altering or adding other rules to the in-path rule list, or by changing the port groups that are used. The default rules allow the following traffic to pass through the Steelhead appliance without attempting optimization:

- **Encrypted Traffic** - Includes HTTPS, SSH, and others.
- **Interactive Traffic** - Includes telnet, ICA, and others.
- **Riverbed Protocols** - Includes the TCP ports used by Riverbed products (that is, the Steelhead appliance, the Interceptor appliance, and the Mobile Controller).

Peering Rules

Peering rules control Steelhead appliance behavior when the appliance detects probe queries.

Peering rules (displayed using the **show in-path peering rules** CLI command) are an ordered list of fields that a Steelhead appliance uses to match with incoming SYN packet fields (for example, source or destination subnet, IP address, VLAN, or TCP port) and with the in-path IP address of the probing Steelhead appliance. If more than one in-path interface exists on the probing Steelhead appliance, apply one peering rule for each in-path interface. Peering rules are especially useful in complex networks.

Peering rule actions are as follows:

- **Pass** - The receiving Steelhead appliance does not respond to the probing Steelhead appliance, and allows the SYN+probe packet to continue through the network.
- **Accept** - The receiving Steelhead appliance responds to the probing Steelhead appliance and becomes the remote-side Steelhead appliance (that is, the peer Steelhead appliance) for the optimized connection.
- **Auto** - If the receiving Steelhead appliance is not using enhanced auto-discovery, this has the same effect as the **Accept** peering rule action. If enhanced auto-discovery is enabled, the Steelhead appliance becomes the optimization peer only if it is the last Steelhead appliance in the path to the server.

If a packet does not match any peering rule in the list, the default rule applies.

The Kickoff and Automatic Kickoff Features

The kickoff feature provides you with a simple way to ensure that unoptimized active TCP connections passing through the Steelhead appliance can be reset. When a connection is reset, it tries to reestablish itself using the SYN, SYN-ACK, ACK handshake. The Steelhead appliance uses its in-path rule table to determine if the connection should be optimized.

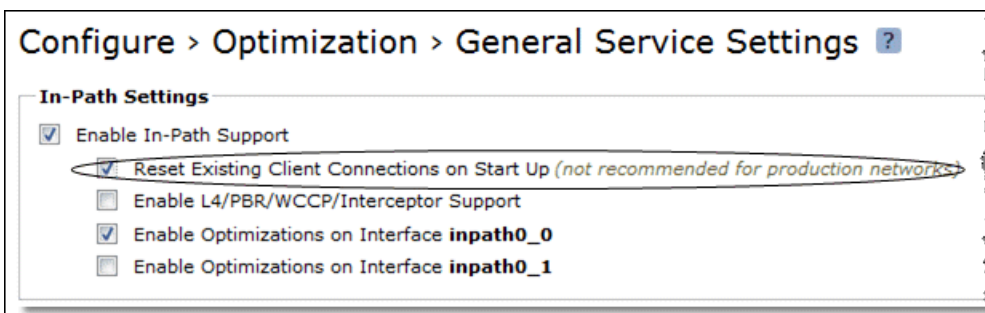
For more information about in-path rules, see [“In-Path Rules” on page 24](#).

Connections can pass through a Steelhead appliance unoptimized when they are set up and active before the Steelhead appliance optimization service is running. By default, the Steelhead appliance does not reset legacy connections and reports them as pre-existing.

The main difference between the auto kickoff feature and the kickoff feature is that kickoff has a global setting that can affect all existing connections passing through a Steelhead appliance. The global setting sends a reset to all connections, regardless of whether they need one. The global setting is not recommended for production networks, but you can use it in lab test scenarios. By default, this setting is not enabled. You can enable this setting in the Management Console, or with the CLI command **in-path kickoff**.

The automatic kickoff feature is included in RiOS v6.1 or later.

Figure 1-5. The Automatic Kickoff Feature Global Setting



Note: You can reset individual connections manually on the Steelhead appliance. This forces an existing connection from optimized to pass-through, or visa-versa, when you test a new rule in the Steelhead appliance's in-path rule table. You can also use manual reset for diagnostic purposes. You can set this feature on the Current Connections page in the Management Console, or with the **tcp connection send pass-reset** commands.

If you configure the automatic kickoff feature, when a Steelhead appliance comes out of bypass mode, it automatically resets (by sending RST to the client and server) only the pre-existing TCP connections that match in-path rules for which automatic kickoff is enabled.

If automatic kickoff and the global kickoff feature are both enabled on the same Steelhead appliance, the global kickoff setting takes precedence.

One of the reasons to use the automatic kickoff feature instead of the kickoff feature is that you can enable it as part of an in-path rule for optimization (Figure 1-6). Automatic kickoff is only available for auto-discovery and fixed-target rules. When you enable automatic kickoff as part of an in-path rule, and the rule matches packet flow for a pre-existing connection, the individual connection is reset automatically. Connections that are pre-existing and do not match an in-path rule with automatic kickoff enabled are unaffected. If you add a new rule with automatic kickoff, any pre-existing connection that matches the new rule is not affected until the next service restart.

Figure 1-6. Enable Automatic Kickoff

The screenshot shows the 'Configure > Optimization > In-Path Rules' configuration page. At the top, there are buttons: 'Add a New In-Path Rule', 'Remove Selected Rules', and 'Move Selected Rules...'. Below these are several configuration fields:

- Type:** Auto Discover (dropdown)
- Source Subnet:** 0.0.0.0/0 (text box)
- Destination Subnet:** 0.0.0.0/0 (text box)
- Port or Port Label:** all (text box)
- VLAN Tag ID:** all (text box)
- Preoptimization Policy:** None (dropdown)
- Latency Optimization Policy:** Normal (dropdown)
- Data Reduction Policy:** Normal (dropdown)
- Auto Kickoff:** ☒ (checkbox, circled in red)
- Neural Framing Mode:** Always (dropdown)
- WAN Visibility Mode:** Correct Addressing (dropdown)
- Position:** End (dropdown)
- Description:** (empty text box)
- Enable Rule:** ☒ (checkbox)

At the bottom left is an 'Add' button.

You use automatic kickoff primarily when you deploy Steelhead appliances in data protection environments. In data protection deployments, connections carrying the data replication traffic between the two storage arrays are often long-lived. This poses a problem if the connections are established as unoptimized or pass-through (for example, if the Steelhead appliance is offline during connection setup), because the connections can remain unoptimized for a long time. Without the automatic kickoff on a Steelhead appliance, you must manually intervene to reset the connections carrying data replication traffic on one of the storage arrays.

For more information about data protection deployment, see [“Data Protection Deployments”](#) on page 339.

Although you can use automatic kickoff for any type of optimizable connection, the majority of connections for office applications—Web, email and so on—are comparatively short-lived and begin to be optimized after a brief period of time without any need for a reset.

When using the automatic kickoff feature, be aware of the following:

- Automatic kickoff does not have a timer.

A pre-existing connection that remains inactive for a period of time is reset as soon as there is packet flow and it matches an in-path rule that has auto kickoff enabled. After the connection has been reset, an internal flag is set to prevent further kick offs for the connection unless the optimization service is restarted.

- Take note when you enable automatic kickoff to make sure you do not cause undesired behavior.

For example, in a design in which there is network asymmetry, if one or more Steelhead appliance neighbors are configured and an in-path rule with automatic kickoff matches the connection, then the connection is kicked off even after detecting only one side of the handshake conversation.

For details on configuring the automatic kickoff feature, see the *Riverbed Command-Line Interface Reference Manual* and *Steelhead Appliance Management Console User's Guide*.

Controlling Optimization Configuration Examples

The following examples show common deployment configurations:

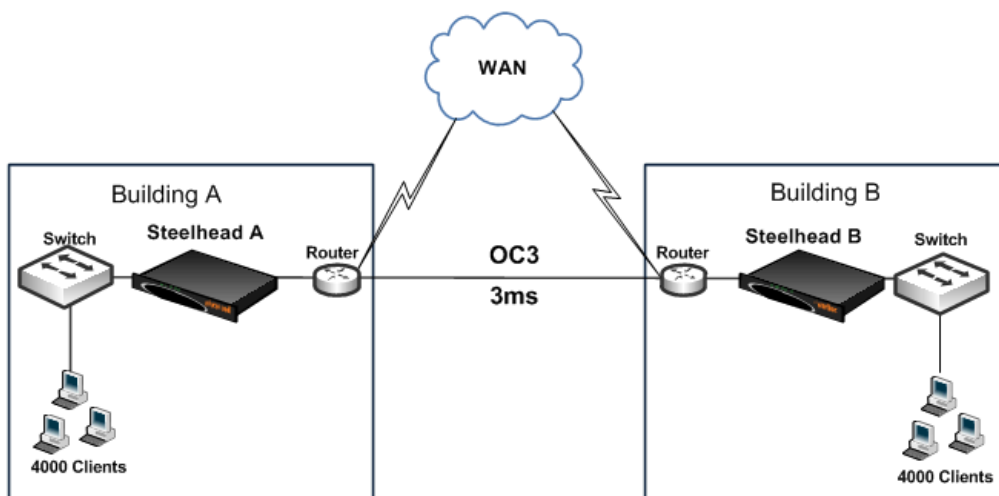
- [“Configuring High-Bandwidth, Low-Latency Environment” on page 28](#)
- [“Configuring Pass-Through Transit Traffic” on page 30](#)

Configuring High-Bandwidth, Low-Latency Environment

To show how in-path and peering rules might be used when designing Steelhead appliance deployments, consider a network that has high bandwidth, low latency, and a large number of users.

[Figure 1-7](#) shows this scenario occurring between two buildings at the same site. In this situation, you want to select Steelhead appliance models to optimize traffic going to and from the WAN. However, you do not want to optimize traffic flowing between Steelhead appliance A and Steelhead appliance B.

Figure 1-7. High Bandwidth Utilization, Low Latency, and Many Connections Between Steelhead Appliances



You can achieve this result as follows:

- **In-path Rules** - You can configure in-path rules on each of the Steelhead appliances (in Building A and Building B) so that the Steelhead appliances do not perform auto-discovery on any of the subnets in Building A and Building B. This option requires knowledge of all subnets within the two buildings, and also requires that you update the list of subnets as the network is modified.
- **Peering Rules** - You can configure peering rules on Steelhead A and Steelhead B that pass through probe packets with in-path IP addresses of the other Steelhead appliance (Steelhead A passes through probe packets with in-path IP addresses of Steelhead B, and vice versa). Using peering rules would require:
 - less initial configuration.
 - less ongoing maintenance, because you do not need to update the list of subnets in the list of peering rules for each of the Steelhead appliances.

Figure 1-8 shows how to use peering rules to prevent optimization from occurring between two Steelhead appliances and still allow optimization for traffic going to and from the WAN.

Figure 1-8. Peering Rules for High Utilization Between Steelhead Appliances

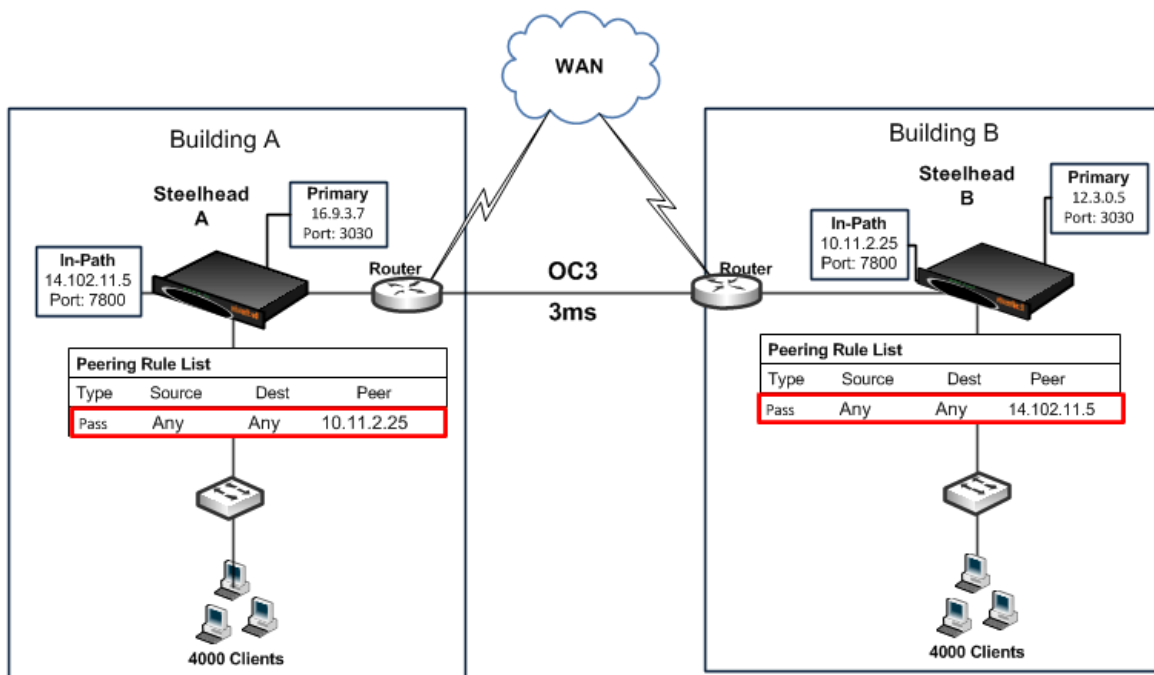


Figure 1-8 shows the following:

- Steelhead A has a **Pass** peering rule for all traffic coming from the Steelhead B in-path interface. When this happens, Steelhead A allows connections from Steelhead B to pass through it unoptimized.
- Steelhead B has a **Pass** peering rule for all traffic coming from the Steelhead A in-path interface. When this happens, Steelhead B allows connections from Steelhead A to pass through it unoptimized.

To configure a high-bandwidth, low-latency environment

1. On Steelhead A, connect to the CLI and enter the following commands:

```
enable
configure terminal
in-path peering rule pass peer 10.11.2.25 rulenum end
```

2. On Steelhead B, connect to the CLI and enter the following commands:

```
enable
configure terminal
in-path peering rule pass peer 14.102.11.5 rulenum end
```

If a packet does not apply to any of the configured peering rules, the auto-peering rule is used.

Configuring Pass-Through Transit Traffic

Transit traffic is data that is flowing through a Steelhead appliance whose source or destination is not local to the Steelhead appliance.

A Steelhead appliance must optimize only traffic that is initiated or terminated at the site where it resides—any extra WAN hops between the Steelhead appliance and the client or server greatly reduces the optimization benefits seen by those connections.

Figure 1-9 shows the Steelhead appliance at the Chicago site detects transit traffic to and from San Francisco and New York (traffic that is not initiated or terminated in Chicago). You want the initiating Steelhead appliance (San Francisco) and the terminating Steelhead appliance (New York) to optimize the connection. You do not want the Steelhead appliance in Chicago to optimize the connection.

Figure 1-9. Transit Traffic

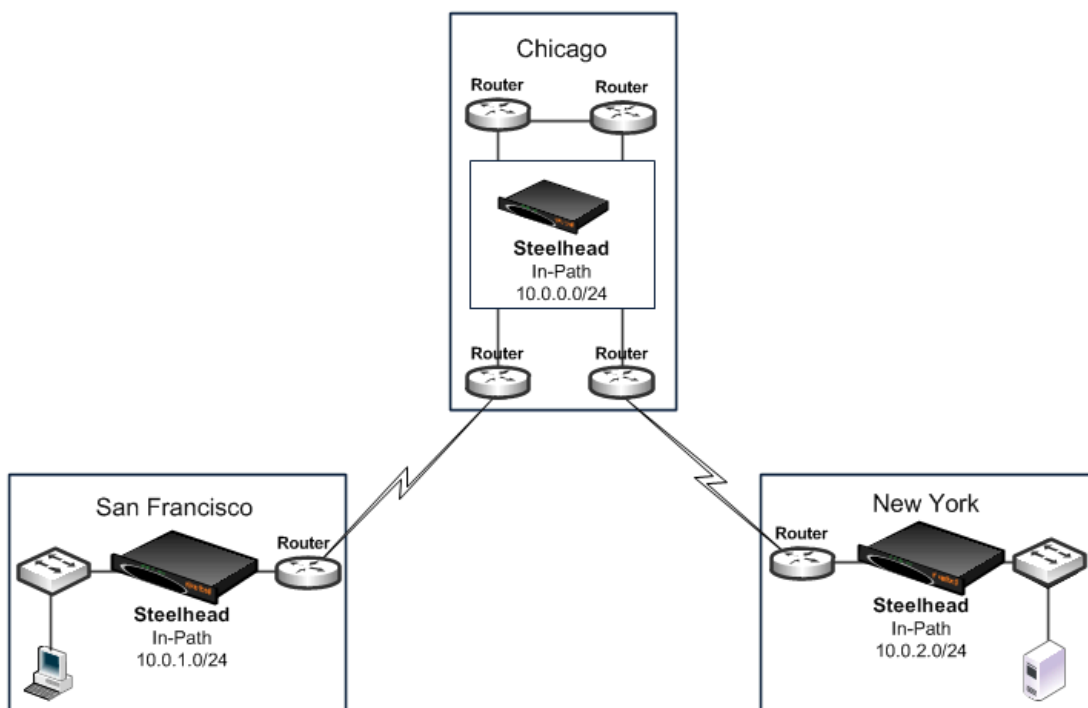


Figure 1-9 shows the possible solutions to resolve this transit traffic issue. These points do not include the configuration for features such as duplex, alarms, and DNS. Also assume that the default in-path rules are configured on all three Steelhead appliances. Because the default action for in-path rules and peering rules is to use auto-discovery, two in-path and two peering rules must be configured on the Chicago Steelhead appliance.

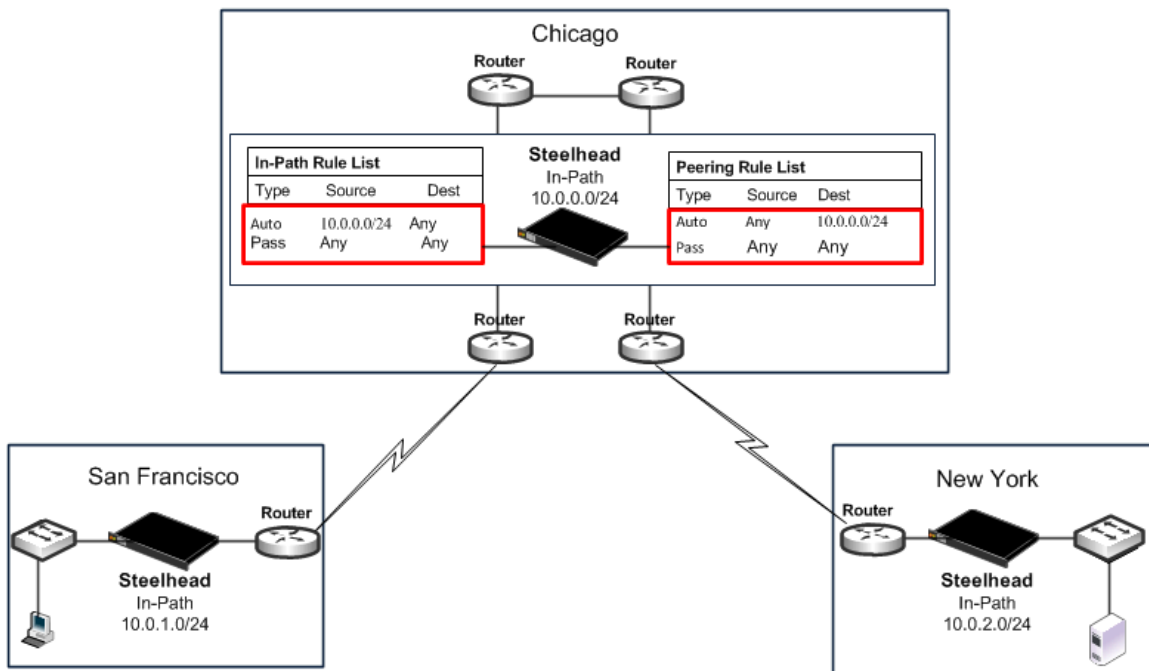
The following scenarios are possible solutions:

- **Manual peering and in-path rules** - Configure in-path and peering rules on the Chicago Steelhead appliance to ignore transit traffic. You can configure peering rules for transit traffic by using the following CLI commands:

```
enable
configure terminal
in-path rule auto srcaddr 10.0.0.0/24 rulenum end
in-path rule pass rulenum end
in-path peering rule auto dest 10.0.0.0/24 rulenum end
in-path peering rule pass rulenum end
```

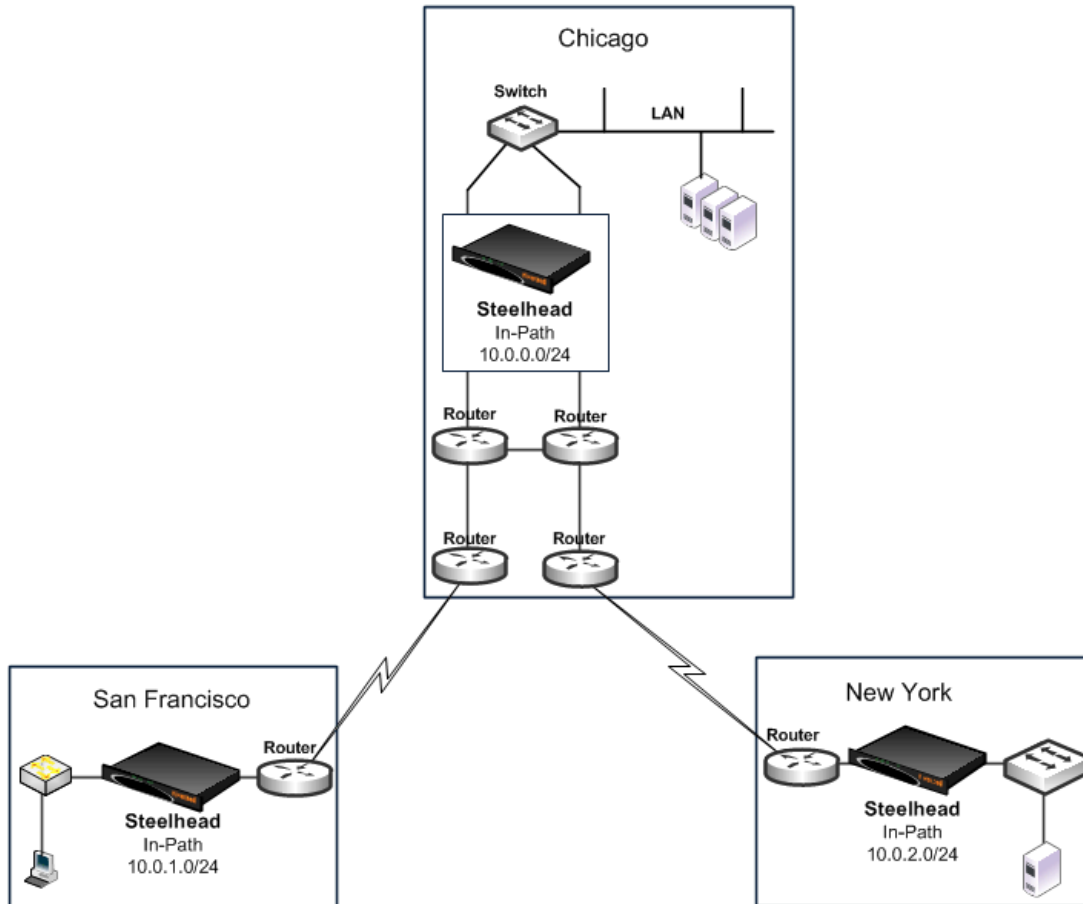
For details on peering rules, see [“Peering Rules” on page 25](#).

Figure 1-10. Peering Rules for Transit Traffic



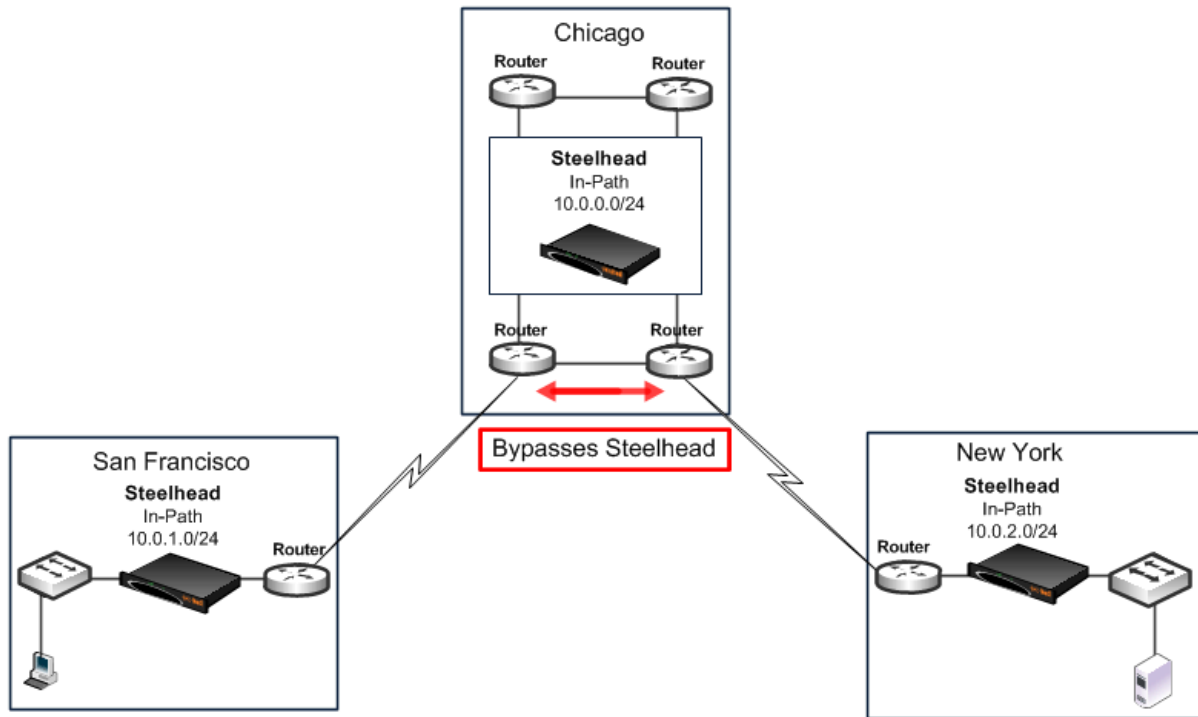
- **Adjust network infrastructure** - Relocate the Chicago Steelhead appliance so that traffic initiated in San Francisco and destined for New York does not pass through the Chicago Steelhead appliance. The Chicago Steelhead appliance only detects traffic that is initiated or terminated at the Chicago site. [Figure 1-11](#) shows relocating the Chicago Steelhead appliance to detect only traffic that is initiated or terminated at the Chicago site.

Figure 1-11. Resolving Transit Traffic by Adjusting Network Infrastructure



- **Adjust traffic flow** - Configure the two routers at the Chicago site to bypass the Chicago Steelhead appliance. [Figure 1-12](#) shows the flow of traffic (initiated or terminated in San Francisco or New York) when the routers at the Chicago site are configured to bypass the Chicago Steelhead appliance.

Figure 1-12. Resolving Transit Traffic by Adjusting Traffic Flow



- **Enable enhanced auto-discovery** - Enable enhanced auto-discovery on all of the Steelhead appliances (in San Francisco, Chicago, and New York). Enhanced auto-discovery enables Steelhead appliances to automatically find the first and the last Steelhead appliance a given that packet must traverse. This ensures that a packet does not become transit traffic. This feature is available in RiOS v4.0.x or later. For details on enhanced auto-discovery, see [“Configuring Enhanced Auto-Discovery” on page 21](#).

Fixed-Target In-Path Rules

A fixed-target in-path rule enables you to manually specify a remote Steelhead appliance for optimization. As with all in-path rules, fixed-target in-path rules are executed only for SYN packets, and therefore are configured on the initiating or client-side Steelhead appliance. This section includes the following topics:

- [“Configuring a Fixed-Target In-Path Rule for an In-Path Deployment” on page 34](#)
- [“Fixed-Target In-Path Rule for an Out-Of-Path Deployment” on page 35](#)

For details on in-path rules, see [“In-Path Rules” on page 24](#).

You can use fixed-target in-path rules in environments where the auto-discovery process cannot work.

A fixed-target rule requires the input of at least one target Steelhead appliance; you can also specify an optional backup Steelhead appliance.

Fixed-target in-path rules have several disadvantages compared to auto-discovery:

- You can not easily determine which subnets to include in the fixed-target rule.

- Ongoing modifications to rules are needed as new subnets or Steelhead appliances are added to the network.
- Currently, you can specify two remote Steelhead appliances. All traffic is directed to the first Steelhead appliance until it reaches capacity, or until it stops responding to requests to connect. Traffic is then directed to the second Steelhead appliance (until it reaches capacity, or until it stops responding to requests to connect).

Note: Because of these disadvantages, fixed-target in-path rules are not as desirable as auto-discovery. In general, use fixed-target rules only when you cannot use auto-discovery.

LAN data flow has a significant difference depending on whether the fixed-target (or backup) IP address specified in the fixed-target in-path rule is for a Steelhead appliance primary interface or its in-path interface.

Configuring a Fixed-Target In-Path Rule for an In-Path Deployment

In environments where the auto-discovery process does not work, use fixed-target in-path rules to target a remote in-path (physical or virtual) Steelhead appliance. Configure the fixed-target in-path rule to target the remote Steelhead appliance in-path interface.

Examples of environments where the auto-discovery process does not work are as follows:

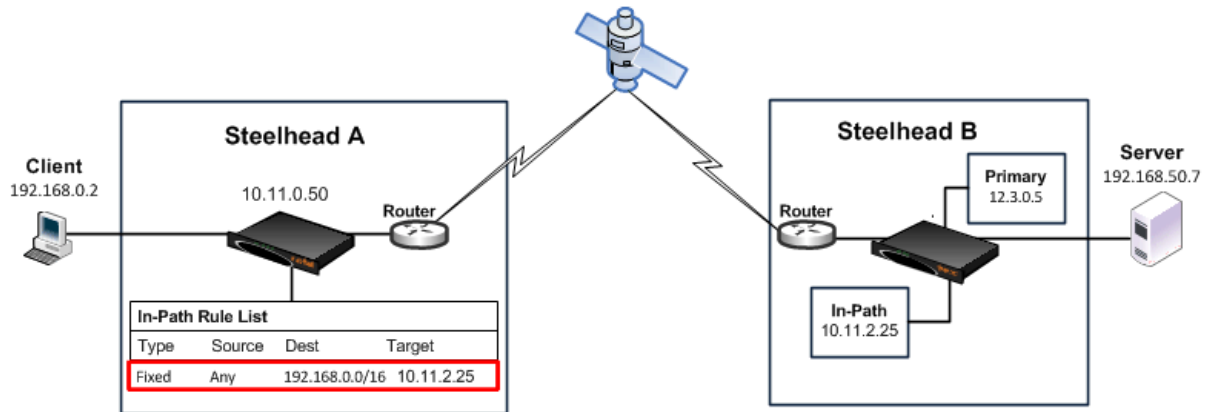
- Traffic traversing the WAN passes through a satellite or other device that strips off TCP options, including those used by auto-discovery.
- Traffic traversing the WAN goes through a device that proxies TCP connections and uses its own TCP connection to transport the traffic. For example, some satellite-based WANs use built-in TCP proxies in their satellite uplinks.

When the target IP address of a fixed-target in-path rule is a Steelhead appliance in-path interface, the traffic between the server-side Steelhead appliance and the server looks like client-to-server traffic; that is, the server detects connections coming from the client IP address. This process is the same as when auto-discovery is used.

[Figure 1-13](#) shows how to use a fixed-target in-path rule to the Steelhead appliance in-path interface. This example uses a fixed-target in-path rule to resolve an issue with a satellite. The satellite gear strips the TCP option from the packet, which means the Steelhead appliance does not detect the TCP option, and the connection cannot be optimized.

To enable the Steelhead appliance to detect the TCP option, you configure a fixed-target in-path rule on the initiating Steelhead appliance (Steelhead A) that targets the terminating Steelhead appliance (Steelhead B) in-path interface.

Figure 1-13. Fixed-Target In-Path Rule to the Steelhead Appliance In-Path Interface



The fixed-target in-path rule specifies that only SYN packets destined for 192.168.0.0/16, Steelhead B subnets, are allowed through to the Site B Steelhead appliance. All other packets are passed through the Steelhead appliance.

You can configure in-path rules using the CLI.

To configure fixed-target in-path rule to an in-path address

- On Steelhead A, connect to the CLI and enter the following commands:

```
enable
configure terminal
in-path rule fixed-target target-addr 10.11.2.25 dstaddr 192.168.0.0/16 rulenum end
```

Fixed-Target In-Path Rule for an Out-Of-Path Deployment

When you enable the remote Steelhead appliance for out-of-path deployment, use fixed-target in-path rules to target the primary IP address. The most important caveat to this deployment method is that traffic to the remote server no longer uses the client IP address. Instead, the server detects connections coming to it from the out-of-path Steelhead appliance primary IP address.

For deployment examples, see, [“Out-Of-Path Deployments” on page 335](#).

Best Practices for Steelhead Appliance Deployments

The following list represents best practices for deploying your Steelhead appliances. These best practices are not requirements, but Riverbed recommends that you follow these suggestions because they lead to designs that require the least amount of initial and ongoing configuration:

- **Use in-path designs** - Whenever possible, use a physical in-path deployment—the most common type of Steelhead appliance deployment. Physical in-path deployments are easier to manage and configure than WCCP, PBR, and Layer-4 designs. In-path designs generally require no extra configuration on the connected routers or switches. If desired, you can limit traffic to be optimized on the Steelhead appliance.

For details, see [“Physical In-Path Deployments” on page 171](#).

- **Use the correct cables** - To ensure that traffic flows not only when the Steelhead appliance is optimizing traffic, but also when the Steelhead appliance transitions to fail-to-wire mode, use the appropriate crossover or straight-through cable to connect the Steelhead appliance to a router or switch. Verify the cable selection by removing power from the Steelhead appliance and then test connectivity through it.

For details, see [“Choosing the Correct Cables” on page 177](#).

- **Set matching duplex speeds** - The most common cause of performance issues is duplex mismatch on the Steelhead appliance WAN or LAN interfaces, or on the interface of a device connected to the Steelhead appliance. Most commonly, the issue with the interface of a network device deployed prior to the Steelhead appliance.

For details on duplex settings, see [“Cabling and Duplex” on page 177](#). For details on troubleshooting duplex mismatch, see [“Physical In-Path Deployments” on page 171](#).

- **Minimize the effect of link state transition** - Use the Cisco **spanning-tree portfast** command on Cisco switches, or similar configuration options on your routers and switches, to minimize the amount of time an interface stops forwarding traffic when the Steelhead appliance transitions to failure mode.

For details, see [“Fail-to-Wire Mode” on page 174](#).

- **Use serial rather than parallel designs** - Parallel designs are physical in-path designs in which a Steelhead appliance has some, but not all, of the WAN links passing through it, and other Steelhead appliances have the remaining WAN links passing through them. Connection forwarding must be configured for parallel designs. In general, it is easier to use physical in-path designs where one Steelhead appliance has all of the links to the WAN passing through it.

For details on serial designs, see [“Physical In-Path Deployments” on page 171](#). For details on connection forwarding, see [“Connection Forwarding” on page 42](#).

- **Do not optimize transit traffic** - Ideally, Steelhead appliances optimize only traffic that is initiated or terminated at its local site. To avoid optimizing transit traffic, deploy the Steelhead appliances where the LAN connects to the WAN and not where LAN-to-LAN or WAN-to-WAN traffic can pass through (or be redirected to) the Steelhead appliance.

For details, see [“Configuring Pass-Through Transit Traffic” on page 30](#).

- **Position your Steelhead appliances close to your network end points** - For optimal performance, minimize latency between Steelhead appliances and their respective clients and servers. By deploying Steelhead appliances as close as possible to your network end points (that is, place client-side Steelhead appliances as close to your clients as possible, and place server-side Steelhead appliances as close to your servers as possible).

- **Use WAN visibility modes that interoperate with monitoring, QoS, and security infrastructure** - RiOS currently supports four different WAN visibility modes, including granular control for their usage. This ensures that the most appropriate mode is used.
For details, see [“WAN Visibility Modes” on page 49](#).
- **Use RiOS data store synchronization** - Regardless of the deployment type or clustering used at a site, RiOS data store synchronization can allow significant bandwidth optimization, even after a Steelhead appliance or hard drive failure.
For details, see [“RiOS Data Store Synchronization” on page 15](#).
- **Use connection forwarding and allow-failure in a WCCP cluster** - In a WCCP cluster, use connection forwarding and the **allow-failure** CLI option between Steelhead appliances. For details, see [“Connection Forwarding” on page 42](#).
- **Avoid using fixed-target in-path rules** - Use the auto-discovery feature whenever possible, thus avoiding the need to define fixed-target, in-path rules.
For details on auto-discovery, see [“The Auto-Discovery Protocol” on page 18](#). For details on fixed-target in-path rules, see [“Fixed-Target In-Path Rules” on page 33](#).
- **Understand in-path rules versus peering rules** - Use in-path rules to modify Steelhead appliance behavior when a connection is *initiated*. Use peering rules to modify Steelhead appliance behavior when it detects auto-discovery tagged packets.
For details, see [“In-Path Rules” on page 24](#) and [“Peering Rules” on page 25](#).
- **Use Riverbed Professional Services or an authorized Riverbed Partner** - Training (both standard and custom) and consultation are available for small to large, and extra-large, deployments.
For details, contact Riverbed Professional Services by email at email proserve@riverbed.com or go to <http://www.riverbed.com/services-training/Services-Training.html>.

CHAPTER 2 Network Integration Tools

This chapter describes Steelhead appliance tools that you can integrate with your network. This chapter includes the following sections:

- [“Redundancy and Clustering” on page 39](#)
- [“Fail-to-Wire and Fail-to-Block” on page 41](#)
- [“Overview of Link State Propagation” on page 41](#)
- [“Connection Forwarding” on page 42](#)
- [“Overview of Simplified Routing” on page 45](#)

Redundancy and Clustering

This section describes redundant deployment of Steelhead appliances in your network. Redundant deployment ensures that optimization continues in case of a Steelhead appliance failure. Redundancy and clustering options are available for each type of deployment. This section includes the following topics:

- [“Physical In-Path Deployments” on page 39](#)
- [“Virtual In-Path Deployments” on page 40](#)
- [“Out-of-Path Deployments” on page 40](#)

Physical In-Path Deployments

The following redundancy options for physical in-path deployments are available:

- **Master and Backup In-Path Deployment** - In a master and backup deployment, two Steelhead appliances are placed in a physical in-path mode. One of the Steelhead appliances is configured as a master, and the other as the backup. The master Steelhead appliance (typically the Steelhead appliance closest to the LAN) optimizes traffic, and the backup Steelhead appliance constantly checks to make sure the master Steelhead appliance is functioning. If the backup Steelhead appliance cannot reach the master, it begins optimizing new connections until the master comes back up. After the master has recovered, the backup Steelhead appliance stops optimizing new connections, and allows the master to resume optimizing. However, the backup Steelhead appliance continues to optimize connections that were made while the master was down. This is the only time, immediately after a recovery from a master failure, that connections are optimized by both the master Steelhead appliance and the backup.

For details, see [“Configuring Master and Backup Deployments” on page 184](#).

- **Serial Cluster In-Path Deployment** - In a serial cluster deployment, two or more Steelhead appliances are placed in a physical in-path mode, and the Steelhead appliances concurrently optimize connections. Because the Steelhead appliance closest to the LAN detects the combined LAN bandwidth of all of the Steelhead appliances in the series, serial clustering is supported on only the higher-end Steelhead appliance models. Serial clustering requires configuring peering rules on the Steelhead appliances to prevent them from choosing each other as optimization peers.

Deployments that use connection forwarding with multiple Steelhead appliances, each covering different links to the WAN, do not necessarily provide redundancy.

For details on serial clustering, see [“Configuring Serial Cluster Deployments” on page 187](#). For details on connection forwarding and multiple Steelhead appliance deployment, see [“Connection Forwarding” on page 42](#) and [“Configuring Multiple WAN Router Deployments with Connection Forwarding” on page 198](#).

Virtual In-Path Deployments

For virtual in-path deployments, the clustering and redundancy options vary depending on which redirection method is being used. WCCP, the most common virtual in-path deployment method, allows options like N+1 redundancy and 1+1 redundancy.

For details on virtual in-path deployments, see [“Virtual In-Path Deployments” on page 217](#).

Out-of-Path Deployments

For an out-of-path deployment, you can configure two Steelhead appliances (a primary and a backup), with fixed-target rules that specify traffic for optimization. If the primary Steelhead appliance becomes unreachable, new connections are optimized by the backup Steelhead appliance. If the backup Steelhead appliance is down, no optimization occurs, and traffic is passed through the network unoptimized.

The master Steelhead appliance uses an Out-of-Band (OOB) connection. The OOB connection is a single, unique TCP connection that communicates internal information only; it does not contain optimized data. If the master Steelhead appliance becomes unavailable, it loses this OOB connection and the OOB connection times out in approximately 40-45 seconds. After the OOB connection times out, the client-side Steelhead appliance declares the master Steelhead appliance unavailable and connects to the backup Steelhead appliance.

During the 40-45 second delay before the client-side Steelhead appliance declares a peer unavailable, it passes through any incoming new connections; they are not blackholed.

Although the client-side Steelhead appliance is using the backup Steelhead appliance for optimization, it attempts to connect to the master Steelhead appliance every 30 seconds. If the connection succeeds, the client-side Steelhead appliance reconnects to the master Steelhead appliance for any new connections. Existing connections remain on the backup Steelhead appliance for their duration. This is the only time, immediately after a recovery from a master failure, that connections are optimized by both the master Steelhead appliance and the backup.


If both the master and backup Steelhead appliances become unreachable, the client-side Steelhead appliance tries to connect to both appliances every 30 seconds. Any new connections are passed through the network unoptimized.

For details on out-of-path deployments, see [“Out-of-Path Deployments” on page 335](#).

Fail-to-Wire and Fail-to-Block

In physical in-path deployments, the Steelhead appliance LAN and WAN ports that traffic flows through are internally connected by circuitry that can take special action in the event of a disk failure, a software crash, a runaway software process, or even loss of power to the Steelhead appliance.

All Steelhead appliance models and in-path network interface cards support fail-to-wire mode, where, in the event of a failure or loss of power, the LAN and WAN ports become internally connected as if they were the ends of a crossover cable, thereby providing uninterrupted transmission of data over the WAN. The default failure mode is fail-to-wire mode.

 VSH supports fail-to-wire or fail-to-block only when deployed with a Riverbed NIC. For more details, see the *Virtual Steelhead Appliance Installation Guide*.


Certain in-path network interface cards also support a fail-to-block mode, where in the event of a failure or loss of power, the Steelhead appliance LAN and WAN interfaces completely lose link status. When fail-to-block is enabled, a failed Steelhead appliance blocks traffic along its path, forcing traffic to be re-routed onto other paths (where the remaining Steelhead appliances are deployed).


For details on fail-to-block mode, see [“Fail-to-Block Mode” on page 175](#). For details on Steelhead appliance LAN and WAN ports and physical in-path deployments, see [“The Logical In-Path Interface” on page 172](#). For details on physical in-path deployments, see [“Physical In-Path Deployments” on page 171](#).

Overview of Link State Propagation

In physical in-path deployments, link state propagation (LSP) can shorten the recovery time of a link failure. Link state propagation communicates link status between the devices connected to the Steelhead appliance. When this feature is enabled, the link state of each Steelhead appliance LAN-WAN pair is monitored. If either physical port loses link status, the other corresponding physical port brings its link down. Allowing link failure to quickly propagate through a chain of devices, LSP is useful in environments where link status is used for fast failure detection.

In RiOS v6.0 or later, link state propagation is enabled by default.

 CSH models do not support LSP.

 VSH running RiOS v8.0.3 with ESXi 5.0 and later using a Riverbed NIC card support LSP.

These VSH configurations do not support LSP:

- VSH models running ESX/ESXi 4.0 or 4.1
- VSH models running Microsoft Hyper-V
- VSH models running RiOS v8.0.2 and earlier

For details on physical in-path deployments, see [“Physical In-Path Deployments” on page 171](#). For more details on LSP, see [“Configuring Link State Propagation” on page 176](#).

Connection Forwarding

For a Steelhead appliance to optimize a TCP connection, it must detect all of the packets for that connection. When you use connection forwarding, multiple Steelhead appliances work together and share information about which connections are being optimized by each. With connection forwarding, the LAN interface forwards and receives connection-forwarding packets. This section includes the following topics:

- [“Configuring Connection Forwarding” on page 43](#)
- [“Multiple-Interface Support Within Connection Forwarding” on page 44](#)
- [“Failure Handling Within Connection Forwarding” on page 44](#)
- [“Connection-Forwarding Neighbor Latency” on page 45](#)

Steelhead appliances that are configured to use connection forwarding with each other are known as *connection-forwarding neighbors*. If a Steelhead appliance detects a packet belonging to a connection that is optimized by a different Steelhead appliance, it forwards it to the correct Steelhead appliance. When a neighbor Steelhead appliance reaches its optimization capacity, that Steelhead appliance stops optimizing new connections but continues to forward packets for TCP connections being optimized by its neighbors.

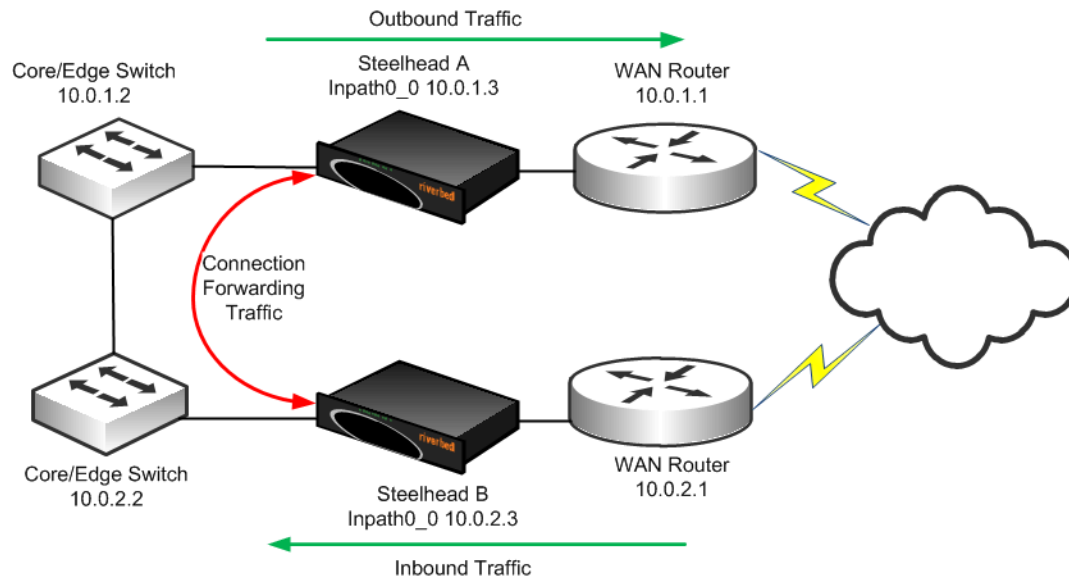
You can use connection forwarding in both physical in-path deployments and virtual in-path deployments. In physical in-path deployments, it is used between Steelhead appliances that are deployed on separate parallel paths to the WAN. In virtual in-path deployments, it is used when the redirection mechanism does not guarantee that packets for a TCP connection are always sent to the same Steelhead appliance. This includes the WCCP protocol, a commonly used virtual in-path deployment method.

Typically, it is easier to design physical in-path deployments that do not require connection forwarding. For example, if you have multiple paths to the WAN, you can use a Steelhead appliance model that supports multiple in-path interfaces, instead of using multiple Steelhead appliances with single in-path interfaces. In general, serial deployments are preferred over parallel deployments.

For details on deployment best practices, see [“Best Practices for Steelhead Appliance Deployments” on page 36](#).

Figure 2-1 shows a site with multiple paths to the WAN. Steelhead A and Steelhead B can be configured as connection-forwarding neighbors. This ensures that if a routing or switching change causes TCP connection packets to change paths, either Steelhead A or Steelhead B can forward the packets back to the correct Steelhead appliance.

Figure 2-1. Connection Forwarding Steelhead Appliances



For information about connection forwarding and MTU sizing, see [“Connection-Forwarding MTU Considerations” on page 447](#).

Configuring Connection Forwarding

The following example is based on the assumption that the Steelhead appliances have already been configured properly for in-path interception.

To configure connection forwarding

1. On Steelhead A, connect to the CLI and enter the following commands:

```
enable
configure terminal
steelhead communication enable
steelhead name SteelheadA main-ip 10.0.2.3
```

2. On Steelhead B, connect to the CLI and enter the following commands:

```
enable
configure terminal
steelhead communication enable
steelhead name SteelheadB main-ip 10.0.1.3
```

When Steelhead A begins optimizing a new TCP connection, it communicates this to Steelhead B, provides the IP addresses and TCP port numbers for the new TCP connection, and defines a dynamic TCP port on which to forward packets.

If Steelhead B detects a packet that matches the connection, it takes the packet, alters its destination IP address to be the in-path IP address of Steelhead A, alters its destination TCP port to be the specific dynamic port that Steelhead A specified for the connection, and transmits the packet using its routing table.

In most environments, Riverbed recommends that you configure connection-forwarding Steelhead appliances to send traffic to each other through the LAN side of the network. Generally, the LAN-side network equipment is connected through low-latency network equipment with more than sufficient connectivity, and the WAN-side equipment might not be directly connected. To make sure that the connection-forwarding neighbor Steelhead appliance sends traffic to each of their in-path IP addresses through the LAN, install a static route for the addresses whose next hop is the LAN gateway device.

For details on connection forwarding in multiple WAN routers, see [“Configuring Basic Connection Forwarding” on page 199](#).

Multiple-Interface Support Within Connection Forwarding

By default, Steelhead appliances communicate with neighbor appliances over a single in-path interface, on whatever is the lowest-numbered, enabled interface. If reachability is lost across the single interface, then the connection-forwarding capabilities is degraded or broken.

The CLI command **steelhead communication multi-interface enable** allows all Steelhead appliance neighbor in-path interface IP addresses to be visible to each peer. This visibility ensures neighbor communication if an interface fails. This command provides a level of interface redundancy; however, you can also think of the multiple-interface option as an improved version of the connection-forwarding protocol. Some additional features, such as the Interceptor appliance load-balancing functions, require you to enable multiple-interface support regardless of the number of interfaces enabled.

Connection-forwarding Steelhead appliances with multiple-interface support attempt to establish communication from every enabled in-path interface to every neighbor appliance in-path interface. Depending on traffic flow, you can forward optimized traffic between Steelhead appliances through any active in-path interfaces. Therefore, in typical environments, Riverbed recommends that all enabled and connected in-path interfaces on the Steelhead appliances be reachable by their connection-forwarding neighbors. Please consult Riverbed Professional Services or account team for environments in which reachability between neighbor in-path interfaces is limited.

Tip: Riverbed recommends that you enable multiple-interface support in all new deployments using connection forwarding. You cannot combine connection-forwarding Steelhead appliances with multiple-interface support enabled with Steelhead appliances without the multiple-interface support enabled.

Failure Handling Within Connection Forwarding

By default, if a Steelhead appliance loses connectivity to a connection-forwarding neighbor, the Steelhead appliance stops attempting to optimize new connections. This behavior can be changed with the **steelhead communication allow-failure** CLI command. If the **allow-failure** command is enabled, a Steelhead appliance continues to optimize new connections, regardless of the state of its neighbors.

For virtual in-path deployments with multiple Steelhead appliances, including WCCP clusters, you must always use connection forwarding and the **allow-failure** command. This is because certain events, such as network failures, and router or Steelhead appliance cluster changes, can cause routers to change the destination Steelhead appliance for TCP connection packets. When this happens, Steelhead appliances must be able to redirect traffic to each other to ensure that optimization continues.

For parallel physical in-path deployments, where multiple paths to the WAN are covered by different Steelhead appliances, connection forwarding is needed because packets for a TCP connection might be routed asymmetrically; that is, the packets for a connection might sometimes go through one path, and other times go through another path. The Steelhead appliances on these paths must use connection forwarding to ensure that the traffic for a TCP connection is always sent to the Steelhead appliance that is performing optimization for that connection.

If the **allow-failure** command is used in a parallel physical in-path deployment, Steelhead appliances optimize only those connections that are routed through the paths with operating Steelhead appliances. TCP connections that are routed across paths without Steelhead appliances (or with a failed Steelhead appliance) are detected by the asymmetric routing detection feature.

For physical in-path deployments, the **allow-failure** command is commonly used with the fail-to-block feature (on supported hardware). When fail-to-block is enabled, a failed Steelhead appliance blocks traffic along its path, forcing traffic to be rerouted onto other paths (where the remaining Steelhead appliances are deployed).

For an example configuration, see [“Configuring Connection Forwarding with Allow-Failure and Fail-to-Block” on page 200](#).

You can configure your Steelhead appliances to automatically detect and report asymmetry within TCP connections as seen by the Steelhead appliance. Asymmetric route auto-detection does not solve asymmetry; it simply detects and reports it, and passes the asymmetric traffic unoptimized. For details on enabling asymmetric route auto-detection, see the *Steelhead Appliance Management Console User’s Guide*.

Connection-Forwarding Neighbor Latency

In general, Riverbed recommends that the maximum round-trip latency between connection forwarding Steelhead appliances is less than one millisecond.

You can deploy Steelhead appliances so that moderate latency exists between connection forwarding Steelhead appliances. Latency has an impact on the optimized traffic because each optimized connection requires communication between all connection forwarding Steelhead appliance neighbors in order to share state about recognizing flows for redirection.

The longest round-trip latency between any two connection forwarding Steelhead appliances should be less than one-fifth of the round-trip latency to the closest optimized remote site. This ensures that connection forwarding communication does not cause connection setup time to be greater for optimized connections compared to unoptimized connections for the closest remote site. Deployments with round-trip latencies higher than 10 milliseconds between connection forwarding Steelhead appliances should only be implemented after a technical consultation with Riverbed.

For more details, see the *Steelhead Appliance Management Console User’s Guide*.

Overview of Simplified Routing

Simplified routing avoids situations when a packet traverses a Steelhead appliance more than once—this is called *packet ricochet*. In environments where the Steelhead appliance is installed in a subnet different than the clients and servers, simplified routing prevents packet ricochet for optimized traffic from the Steelhead appliance.

Figure 2-2 shows an example of packet ricochet when the Steelhead appliance default gateway is configured for the WAN router, the host sits on a different network than the Steelhead appliance, and simplified routing is not enabled.

Figure 2-2. Packet Ricochet when the Steelhead Appliance Default Gateway is on the WAN

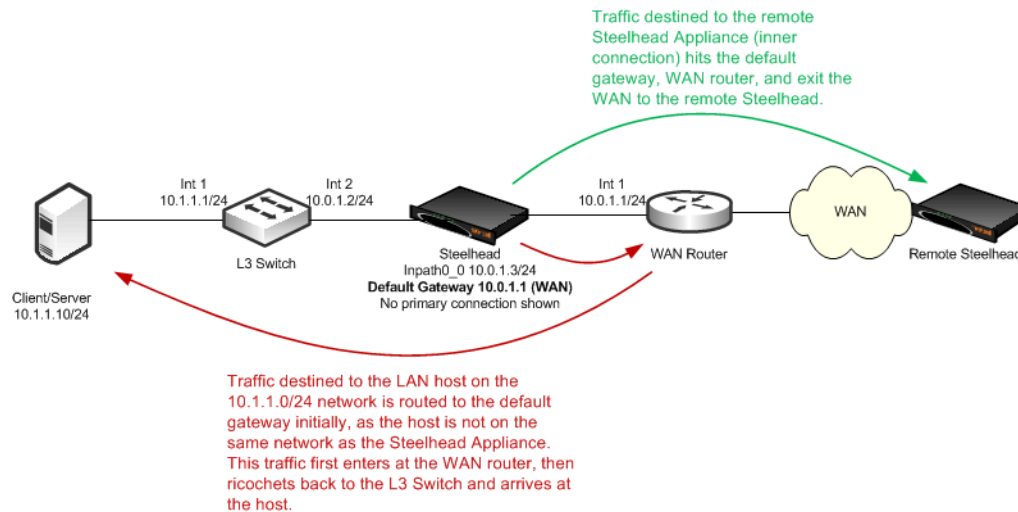
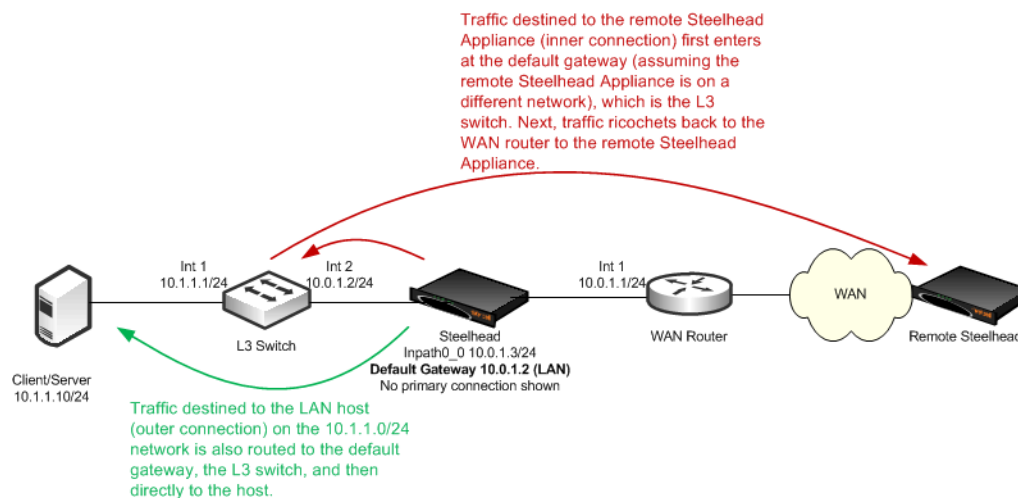


Figure 2-3 shows a similar packet ricochet scenario, but with the default gateway of the Steelhead appliance pointed to the LAN L3 switch.

Figure 2-3. Packet Ricochet when the Steelhead Appliance Default Gateway is on the LAN



In both Figure 2-2 and Figure 2-3, packets for some traffic take a sub-optimal first hop from the Steelhead appliance. While typically the detrimental effects of an extra hop is minor, packet ricochet causes problems in the following environments:

- Some environments that include firewalls or routers with ACLs might not permit traffic to ricochet, or traverse back out the same interface as it came in.
- Some monitoring tools that rely on Netflow or SNMP data count the ricocheted traffic as additional traffic.
- Packet ricochet causes the adjacent network devices to perform unnecessary work.

The packet ricochet scenarios only occur in physically in-path environments where the Steelhead appliance is installed in a subnet different than the clients or servers. In these environments, you can avoid packet ricochet by either configuring static routes or by using simplified routing.

For example, [Figure 2-2](#) shows you can configure a static route for the host network, 10.1.1.0/24 to point directly to the 10.0.1.2 L3 switch, preventing this traffic from using the default gateway. However, the static route method often becomes administratively burdensome, especially in larger or complex LAN environments.

Simplified routing resolves packet ricochet, without using static routes or routing protocols, by building an IP to next-hop MAC address mapping learned from received packets. The Steelhead appliance learns the correct MAC address by examining the packet's destination or source IP and MAC address.

Using [Figure 2-2](#) as an example, assume simplified routing is enabled. If an auto-discovery packet arrives from the WAN to the 10.1.1.10 host, the Steelhead appliance detects the packet with the destination IP of 10.1.1.10 along with the destination MAC of the L3 switch, and records the IP with associated MAC in its simplified routing table—also referred to as the *macmap table*. Whenever the Steelhead appliance generates traffic destined to the 10.1.1.10 host, it uses the associated MAC of the L3 switch instead of the default gateway. This avoids the packet ricochet.

Only use simplified routing for optimized traffic generated by the Steelhead appliance, not pass-through traffic. For pass-through traffic, the Steelhead appliance sends the packets out the opposite WAN or LAN interface as it came in. You can also use simplified routing when the destination IP is on a different subnet than the Steelhead appliance in-path IP. If the destination IP resides on the same network, the Steelhead appliance uses ARP for the correct MAC address. When the destination IP resides on a different network, then a simplified routing entry (if recorded) takes precedence over the default gateway, or by default, any configured static routes. To override the default behavior and have the static routes take precedence over simplified routing, use the following CLI command:

```
in-path simplified mac-def-gw-only
```

Simplified routing plays an important role in maintaining VLAN ID when transmitting across the WAN when the Steelhead appliance is deployed on an 802.1Q trunk and using the full address transparency WAN visibility mode.

For more information about simplified routing in physical in-path deployments, see [“Configuring Simplified Routing” on page 191](#).

CHAPTER 3 WAN Visibility Modes

This chapter describes Steelhead appliance WAN visibility modes and how to configure them. This chapter includes the following sections:

- [“Overview of WAN Visibility” on page 49](#)
- [“Correct Addressing” on page 50](#)
- [“Transparent Addressing” on page 51](#)
- [“Implications of Transparent Addressing” on page 56](#)
- [“The Out-of-Band Connection” on page 69](#)
- [“Configuring WAN Visibility Modes” on page 72](#)

For details on the factors you must consider before you design and deploy the Steelhead appliance in a network environment, see [“Choosing the Right Steelhead Appliance” on page 16](#).

Overview of WAN Visibility

Each LAN-side TCP connection that is optimized by a Steelhead appliance is carried on a unique WAN-side connection. By configuring a WAN visibility mode for some or all optimized connections, you can control which IP addresses and TCP ports are used on these WAN-side TCP connections.

RiOS v6.0 or later offers the following options for configuring WAN visibility modes:

- **Correct Addressing** - WAN-side connections use Steelhead appliance IP addresses and Steelhead appliance server ports.
- **Transparent Addressing** - the following are Transparent Addressing options:
 - **Port Transparency** - WAN-side connections use Steelhead appliance IP addresses but use TCP server ports that mirror the LAN-side connection.
 - **Full Transparency** - WAN-side connections mirror all IP addresses and TCP ports used on the LAN-side connection.
 - **Full Transparency with Forward Reset** - The same as Full Transparency, with an additional packet during auto-discovery to aid with integration of stateful network devices on the WAN.

The most suitable WAN visibility mode depends primarily on your existing network configuration. For example, if you manage IP address-based or TCP port-based QoS policies for optimized traffic on your WAN or WAN routers, you might use full address transparency or port transparency. However, if you need your optimized traffic to pass through a content-scanning firewall that creates alarms when application ports are used on optimized traffic payload, you might use correct addressing instead. You can configure WAN visibility modes on the client-side Steelhead appliance (where the connection is initiated).

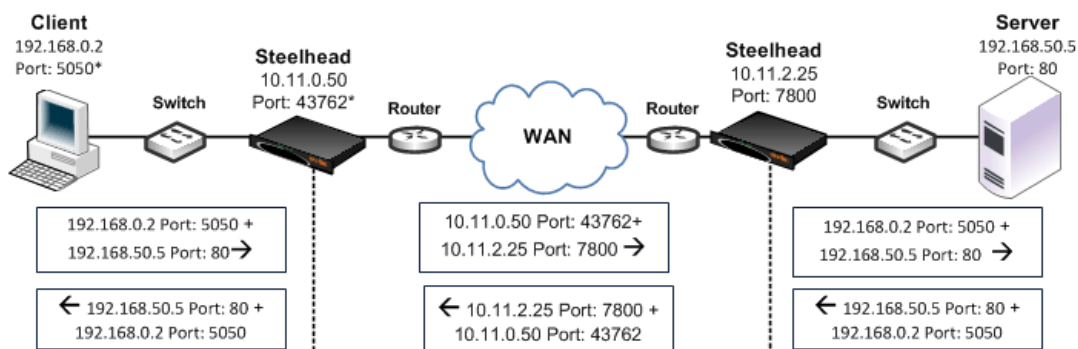
There can be different types of addressing modes on the same Steelhead appliance. Choose the most appropriate addressing mode for your configuration, based on IP addresses, subnets, TCP ports, and VLAN.

Correct Addressing

Correct addressing uses Steelhead appliance IP addresses and port numbers in the TCP/IP packet header fields for optimized traffic in both directions across the WAN. By default, Steelhead appliances use correct addressing.

Figure 3-1 shows TCP/IP packet headers when correct addressing is used. The IP addresses and port numbers of your Steelhead appliances are visible across your WAN. Refer to Figure 3-1 to compare it to port transparency (Figure 3-2) and full address transparency (Figure 3-3) packet headers.

Figure 3-1. Correct Addressing



*In a production environment this port number is random for each connection.

Correct addressing uses the following values in your TCP/IP packet headers in both directions:

- **Client to client-side Steelhead appliance** - client IP address and port + Server IP address and port
- **Client-side Steelhead appliance to server-side Steelhead appliance** - client-side Steelhead appliance IP address and port + Server-side Steelhead appliance IP address and port
- **Server-side Steelhead appliance to server** - client IP address and port + Server IP address and port

Correct addressing avoids networking risks that are inherent to enabling transparent addressing.

For details on configuring correct addressing, see [“Configuring WAN Visibility Modes” on page 72](#). For details on avoiding network risks, see [“Implications of Transparent Addressing” on page 56](#).

Correct addressing enables you to use the connection pooling optimization feature. Connection pooling works only for connections optimized using correct addressing. Connection pooling enables Steelhead appliances to create several TCP connections between each other before they are needed. When transparent addressing is enabled, Steelhead appliances cannot create the TCP connections in advance because they cannot detect what types of client and server IP addresses and ports are needed.

For details on connection pooling, see [“Connection Pooling” on page 12](#).

Transparent Addressing

This section describes port transparency and full address transparency. This section includes the following topics:

- [“Port Transparency” on page 52](#)
- [“Full Address Transparency” on page 53](#)
- [“Full Address Transparency with Forward Reset” on page 55](#)

Transparent addressing reuses client and server addressing for optimized traffic across the WAN. Traffic is optimized, although addressing appears to be unchanged. Both optimized and pass-through traffic present identical addressing information to the router and network monitoring devices.

In RiOS v5.0.x or later, transparent addressing can be used in conjunction with many deployment configurations and features, including, but not limited to:

- physical in-path deployments (serial clusters, master/backup, and deployments using connection forwarding).
- virtual in-path deployments (WCCP, PBR, Layer 4 switching, and Interceptor appliance deployments).
- auto-discovery, including enhanced auto-discovery.
- asymmetric route detection.
- QoS marking and classification.
- flow data export.

Transparent addressing does not support the following deployment configurations:

- Server-side out-of-path Steelhead appliance configurations
- Fixed-target rules
- Connection pooling

You configure transparent addressing on the client-side Steelhead appliance (where the connection is initiated). Both the server-side and the client-side Steelhead appliances must support transparent addressing (RiOS v5.0.x or later) for transparent addressing to work. You can configure a Steelhead appliance for transparent addressing even if its peer does not support it. The connection is optimized, but it is not transparent.

When you use full or port transparency, Steelhead appliances add a TCP option field to the packet headers of optimized traffic. This TCP option field is sent between the Steelhead appliances. For transparency to work, this option must not be stripped off by intermediate network devices.

A given pair of Steelhead appliances can also have multiple types of transparent addressing enabled for different connections. For example, a pair of Steelhead appliances can use correct addressing for connections to a destination subnet, and use full address transparency or port transparency for connections to another destination subnet. A pair of Steelhead appliances can also use correct addressing for connections to a destination port, and use full address transparency or port transparency for connections to another destination subnet.

If both port transparency and full address transparency are acceptable solutions, port transparency is preferable. Port transparency avoids potential networking risks that are inherent in enabling full address transparency.

For details, see [“Implications of Transparent Addressing” on page 56](#).

Port Transparency

Port transparency preserves your server port numbers in the TCP/IP header fields for optimized traffic in both directions across the WAN. Traffic is optimized even though the server port number in the TCP/IP header field appears to be unchanged. Routers and network monitoring devices deployed in the WAN segment between the communicating Steelhead appliances can view these preserved fields.

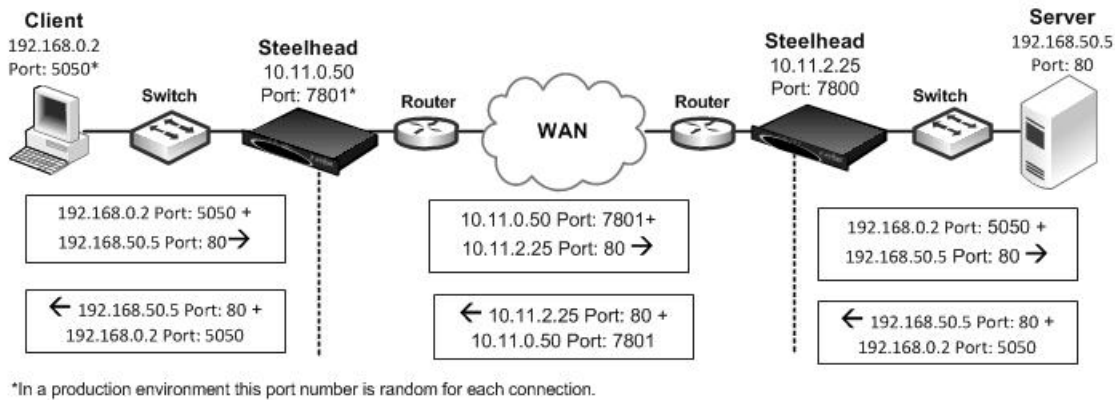
Port transparency does not require dedicated port configurations on your Steelhead appliances.

Port transparency provides server only port visibility. Port transparency does not provide client and server IP address visibility, nor does it provide client port visibility.

[Figure 3-2](#) shows TCP/IP packet headers when port transparency is enabled. Server port numbers are visible across your WAN.

To compare port transparency packet headers to correct addressing packet headers, see [Figure 3-1](#).

Figure 3-2. Port Transparency



Port transparency uses the following values in your TCP/IP packet headers in both directions:

- **Client to client-side Steelhead appliance** - Client IP address and port + server IP address and port.
- **Client-side Steelhead appliance to server-side Steelhead appliance** - Client-side Steelhead appliance IP address and port + server-side Steelhead appliance IP address with server port.
- **Server-side Steelhead appliance to server** - Client IP address and port + server IP address and port.

Use port transparency if you want to manage and enforce QoS policies that are based on destination ports. If your WAN router is following traffic classification rules that are written in terms of TCP destination port numbers, port transparency enables your routers to use existing rules to classify the traffic without any changes.

Port transparency enables network analyzers deployed within the WAN (between the Steelhead appliances) to monitor network activity, and to capture statistics for reporting, by inspecting traffic according to its original TCP destination port number.

For details on configuring port transparency, see [“Configuring WAN Visibility Modes” on page 72](#).

Full Address Transparency

This section describes full address transparency. This section includes the following topics:

- [“Overview of Full Address Transparency” on page 53](#)
- [“Configuring VLANs and Full Address Transparency” on page 54](#)
- [“The Out-of-Band Connection” on page 69](#)

Overview of Full Address Transparency

Full address transparency preserves your client and server IP addresses and port numbers in the TCP/IP header fields for optimized traffic in both directions across the WAN. VLAN tags can also be preserved. Traffic is optimized even though these TCP/IP header fields appear to be unchanged. Routers and network monitoring devices deployed in the WAN segment between the communicating Steelhead appliances can view these preserved fields.

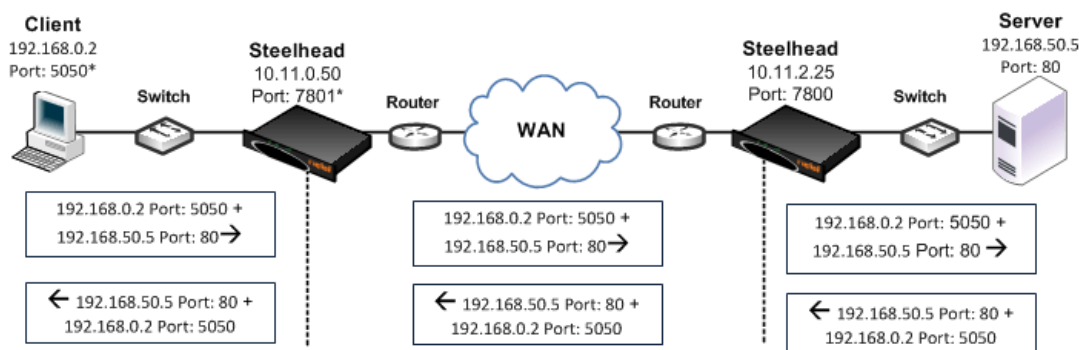
[Figure 3-3](#) shows how TCP/IP packet headers might be addressed when full address transparency is enabled. In this example, Steelhead appliance IP addresses and port numbers are no longer visible on the optimized connections. Client and server IP addresses and port numbers are now visible in both directions across the WAN.

When you enable full address transparency, you have several addressing options for the out-of-band (OOB) connection. The type of addressing you configure for your OOB connection ultimately determines whether the Steelhead appliance in-path IP addresses are used in the TCP/IP packet headers.

For details on OOB, see [“The Out-of-Band Connection” on page 69](#).

To compare full address transparency packet headers with correct addressing packet headers, see [Figure 3-1](#).

Figure 3-3. Full Address Transparency



*In a production environment this port number is random for each connection.

In this example, full address transparency uses the following values in the TCP/IP packet headers in both directions:

- **Client to client-side Steelhead appliance** - Client IP address and port + Server IP address and port.
- **Client-side Steelhead appliance to server-side Steelhead appliance** - Client IP address and port + Server IP address and port.
- **Server-side Steelhead appliance to server** - Client IP address and port + server IP address and port.

For details on configuring full address transparency, see [“Configuring WAN Visibility Modes” on page 72](#).

If both port transparency and full address transparency are acceptable solutions, port transparency is preferable. Port transparency mitigates potential networking risks that are inherent in enabling full address transparency.

For details, see [“Implications of Transparent Addressing” on page 56](#).

However, if you must use your client or server IP addresses across your WAN, full address transparency is your only configuration option. Full address transparency enables network monitoring applications deployed within the WAN (between the Steelhead appliances) to measure traffic sent to the WAN by the end-host. Network routers can also perform load-balancing and policy-based routing. Full address transparency also enables you to manage and enforce QoS policies based on port numbers or IP addresses.

Important: When full address transparency is enabled, router QoS policies cannot distinguish between optimized and unoptimized traffic, even though an optimized packet might represent much more data.

Full address transparency also enables the use of Network Address Translation (NAT). With correct addressing, Steelhead appliances use their own IP addresses in the packet header, which NAT does not recognize. When full address transparency is enabled, the original client and server IP addresses are used, and the connections are recognizable to NAT. However, the type of addressing you configure for your out-of-band (OOB) connection ultimately determines whether the Steelhead appliance in-path IP addresses are used in the TCP/IP packet headers.

Full address transparency also supports several transparency options for the OOB connection.

For details on OOB, see [“The Out-of-Band Connection” on page 69](#).

Important: Some firewalls, QoS devices, and other stateful devices might require additional configuration to successfully allow full transparency connections to optimize. Search the Riverbed Knowledge Base for information about any particular device.

Configuring VLANs and Full Address Transparency

Full address transparency supports transparent VLANs. You can configure full address transparency so that optimized traffic remains on the original VLANs. Because you can keep traffic on the original VLANs, full address transparency enables you to perform VLAN-based QoS on the WAN-side of the Steelhead appliance.

Note: You must first configure WAN visibility full address transparency for VLAN transparency to function correctly.

To configure full address transparency for a VLAN

- On the Steelhead appliance, connect to the CLI and enter the following commands:

```
enable
configure terminal
in-path peering auto
in-path simplified routing all
in-path vlan-conn-based
in-path mac-match-vlan
no in-path probe-caching enable
in-path probe-ftp-data
```



```

in-path probe-mapi-data
write memory
restart

```

Note: You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

If packets on your network use two different VLANs in the forward and reverse directions, see the following Riverbed Knowledge Base article, *Understanding VLANs and Transparency*, located at <https://supportkb.riverbed.com/support/index?page=content&id=S15176>.

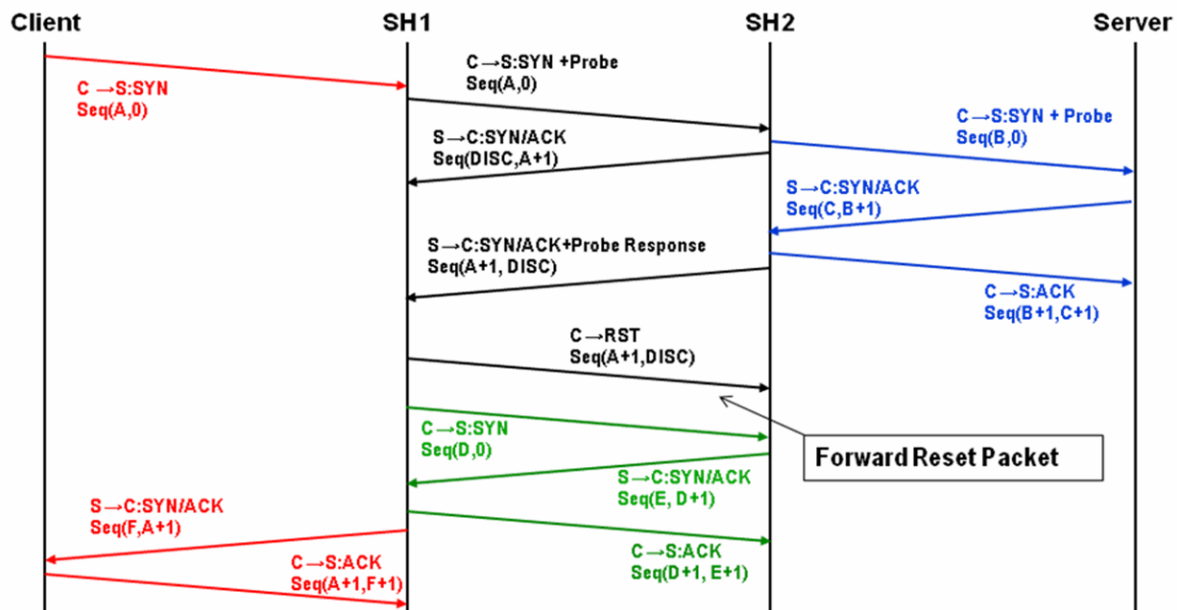
Full Address Transparency with Forward Reset

The *Full Address Transparency with Forward Reset* mode is similar to the *Full Address Transparency WAN visibility* mode. Like Full Address Transparency, use this mode to preserve the client and server IP addresses and TCP ports used for the WAN-side TCP connections between Steelhead appliances. The difference between the two modes happens during the auto-discovery phase, during which TCP reset packets are transmitted on the WAN. These packets help network devices like stateful firewalls separate TCP state between the Steelhead appliances discovery phase and the data transmission phase.

Except for the TCP reset packets during the discovery phase, there are no other differences between Full Address Transparency and Full Address Transparency with Forward Reset, including the addressing of the WAN-side TCP connection, considerations for 802.1Q VLAN tracking, and OOB transparency.

Figure 3-4 shows the auto-discovery packet flow when using the Full Transparency with Forward Reset mode and enhanced auto-discovery. The packet marked *Forward Reset* is the only difference between this mode and the Full Transparency mode.

Figure 3-4. Full Transparency with Forward Reset



In Full Address Transparency with Forward Reset mode, TCP reset packets are transmitted by the initiating Steelhead appliance immediately after the remote Steelhead appliance is discovered. The reset packets traverse the WAN and are absorbed by the remote Steelhead appliance. The packets aid any TCP-aware device on the WAN to understand that the sequence numbers used during the auto-discovery phase is different from the sequence numbers used during the data transmission phase. Example devices include:

- firewalls or other network security devices on the WAN that statefully track TCP sessions, and devices that might block WAN-side Steelhead appliance connections from being created due to seeing different sequence numbers in use.
- QoS devices that alter TCP headers to affect congestion. Examples include Blue Coat Packetshaper appliances using rate policies, and the Allot Netenforcer.

Important: Some firewalls, QoS devices, and other stateful devices might require additional configuration to successfully optimize connections using the full transparency with forward reset connections to operate. Search the Riverbed Knowledge Base for information about any particular device.

Implications of Transparent Addressing

This section describes some of the common problems that are inherent to transparent addressing:

- [“Stateful Systems” on page 56](#)
- [“Network Design Issues” on page 57](#)
- [“Integration into Networks Using NAT” on page 60](#)

The problems described in this section occur with all proxy-based solutions.

Stateful Systems

Transparent addressing can have an impact on systems that monitor or alter the state of TCP connections on the WAN for reporting, security, or congestion control. For example, some stateful firewalls might detect the difference in sequence numbers between the auto-discovery phase and the data transmission phase of fully transparent connections, and react by raising alarms or disallowing connections between the Steelhead appliances. Using the Full Transparency with Forward Reset mode might alleviate this issue, but might also cause monitoring systems to record more TCP connection activity across the WAN than is actually present.

Transparent addressing also does not work with intrusion detection and prevention systems that perform stateful packet inspection. Steelhead appliances use a proprietary Riverbed application protocol to communicate. When intrusion detection and prevention systems perform stateful packet inspections, they expect an application protocol based on the port numbers of the original client and server connection. When these systems discover the Riverbed proprietary application protocol, they perceive this as a mismatch, and log the packet, drop it, trigger an alarm, or all of the above.

You can avoid these problems with stateful systems, which are inherent to transparent addressing, by using correct addressing.

Network Design Issues

This section describes some of the common networking problems that are inherent to transparent addressing. This section includes the following topics:

- [“Network Asymmetry” on page 57](#)
- [“Misrouting Optimized Traffic” on page 58](#)
- [“Firewalls Located Between Steelhead Appliances” on page 59](#)

Network Asymmetry

Enabling full address transparency increases the likelihood of problems inherent to asymmetric routing.

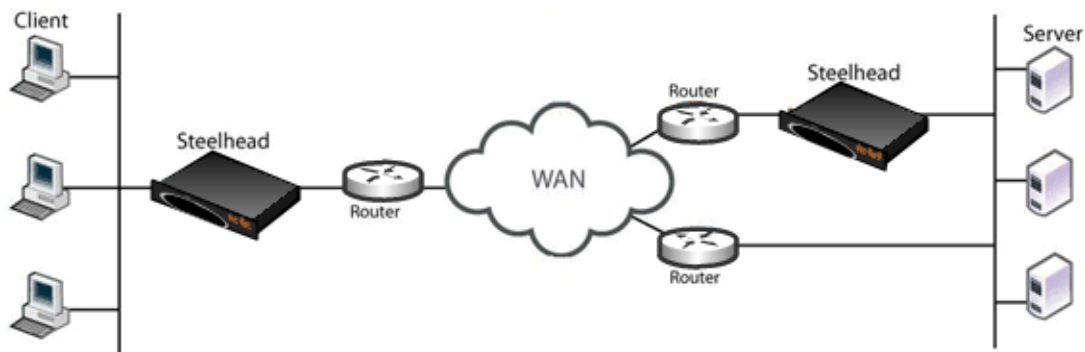
For a connection to be optimized, its packets to and from its LAN hosts must pass through either:

- one or more in-path interfaces on the same Steelhead appliance, or
- one or more in-path interfaces on Steelhead appliances that are configured as connection-forwarding neighbors.

When full address transparency is used, WAN-side routers detect the client or server addresses in the optimized connections packets, and use those address to make routing decisions. If the router has a route to the client or server that does not pass through a Steelhead appliance, and it transmits the optimized packets on that route, the optimized and LAN-side connections might fail.

[Figure 3-5](#) shows a network in which a link to the server location does not have a Steelhead appliance installed. Depending on the exact routing configuration in the [Figure 3-5](#), it is possible that correct addressing can work, but full transparency does not, because the optimized traffic from the client side might be sent through the link that does not have the Steelhead appliance.

Figure 3-5. Server-Side Asymmetric Network



To ensure that all required traffic is optimized and accelerated, you must install a Steelhead appliance on every possible path that a packet traverses. Connection forwarding must also be configured and enabled for each Steelhead appliance.

If there is a path that does not have a Steelhead appliance, it is possible that some traffic will not be optimized.

You can avoid this type of asymmetric routing problem, which is inherent to transparent addressing, by using correct addressing.

For details on connection forwarding, see [“Connection Forwarding” on page 42](#). For details on how to eliminate asymmetric routing problems, see [“Troubleshooting Steelhead Appliance Deployment Problems” on page 431](#).

Note: With RiOS v3.0.x or later, you can configure your Steelhead appliances to automatically detect and report asymmetric routes within your network. For details, see the *Steelhead Appliance Management Console User's Guide*.

Misrouting Optimized Traffic

Enabling transparent addressing introduces the likelihood of misrouting optimized traffic in the event of a Steelhead appliance failure.

Steelhead appliances use a proprietary Riverbed protocol to communicate. Normally, a functioning server-side Steelhead appliance receives a packet from the WAN, and converts the packet to its native format before forwarding it to the server.

In an environment in which transparent addressing is used, if the server-side Steelhead appliance is not functioning, or if a packet is routed along an alternative network path, the packet might go from the client-side Steelhead appliance directly to the server. Because the server-side Steelhead appliance does not have an opportunity to convert the packet to its native format, the server cannot recognize it, and the connection fails.

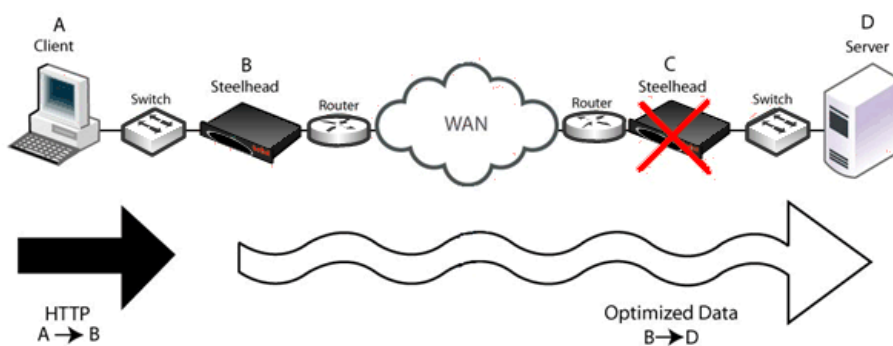
In most cases, the server is able to detect whether a packet contains invalid payload information or, in this case, has an unrecognizable format, and rejects the packet. Assuming that the server does detect that the format is unrecognizable, the server rejects the packet and resets the TCP connection. If the client TCP connection is reset, the client can reconnect to the server without any Steelhead appliance involvement.

This type of traffic misrouting can occur in both directions across the WAN. If the client-side Steelhead appliance experiences a failure, or if an alternate network path exists from the server to the client, traffic might go from the server-side Steelhead appliance directly to the client.

Important: Before enabling and using full address transparency, carefully consider the risks and exposures in the event that a server accepts and routes a packet that has an unrecognizable format.

[Figure 3-6](#) shows traffic being misrouted when the server-side Steelhead appliance fails on a network using transparent addressing.

Figure 3-6. Transparent Addressing and Misrouting Optimized Traffic



The failure scenario follows these steps:

1. Client A sends HTTP data to the server.
2. Steelhead B receives the HTTP data optimizes it. Steelhead B eventually transmits packets carrying the optimized data toward Steelhead C, but due to the transparent addressing mode, the packets are addressed to Server D.
3. Steelhead C suffers a failure and is in fail-to-wire mode, so all packets are traversing it, including the packets from Steelhead B.
4. Server D receives the packets from Steelhead B, but does not recognize the packet format, so the connection fails or suffers an application-dependent error.

You can avoid this type of misrouting problem, which is inherent to transparent addressing, by using correct addressing.

If correct addressing is configured for this scenario, the client-side Steelhead appliance detects that the server-side Steelhead appliance has failed. The client-side Steelhead appliance *automatically* resets the client connection, allowing the client to connect directly to the server without Steelhead appliance involvement.

Firewalls Located Between Steelhead Appliances

If your firewall inspects traffic between two Steelhead appliances, there are addressing issues that you must be aware of.

Figure 3-7. Firewalls and Transparent Addressing



The following table summarizes configuration issues that might arise when a firewall inspects traffic between two Steelhead appliances. Firewall behavior differs, depending on the type of addressing being used. A Yes value indicates that your firewall will perform as expected.

Firewall Configuration	Full Address Transparency	Port Transparency	Correct Addressing
Firewall rules based on a server port	Yes	Yes Note: This configuration does not support active FTP.	Yes, if the following conditions are true: <ul style="list-style-type: none"> • The firewall checks on the session establishment. • The firewall is enabled. • The firewall allows port 7800 traffic.
Firewall rules based on IP addresses	Yes	Yes, if the following conditions are true: <ul style="list-style-type: none"> • IP-based rules are based only on server addresses. • Probe caching is disabled. 	Yes, if the following conditions are true: <ul style="list-style-type: none"> • The firewall checks on the session establishment. • The firewall is enabled. • Probe caching is disabled. For details on disabling probe caching, see the <i>Riverbed Command-Line Interface Reference Manual</i>.

For details on stateful firewalls and intrusion detection and prevention systems, see [“Stateful Systems” on page 56](#).

Integration into Networks Using NAT

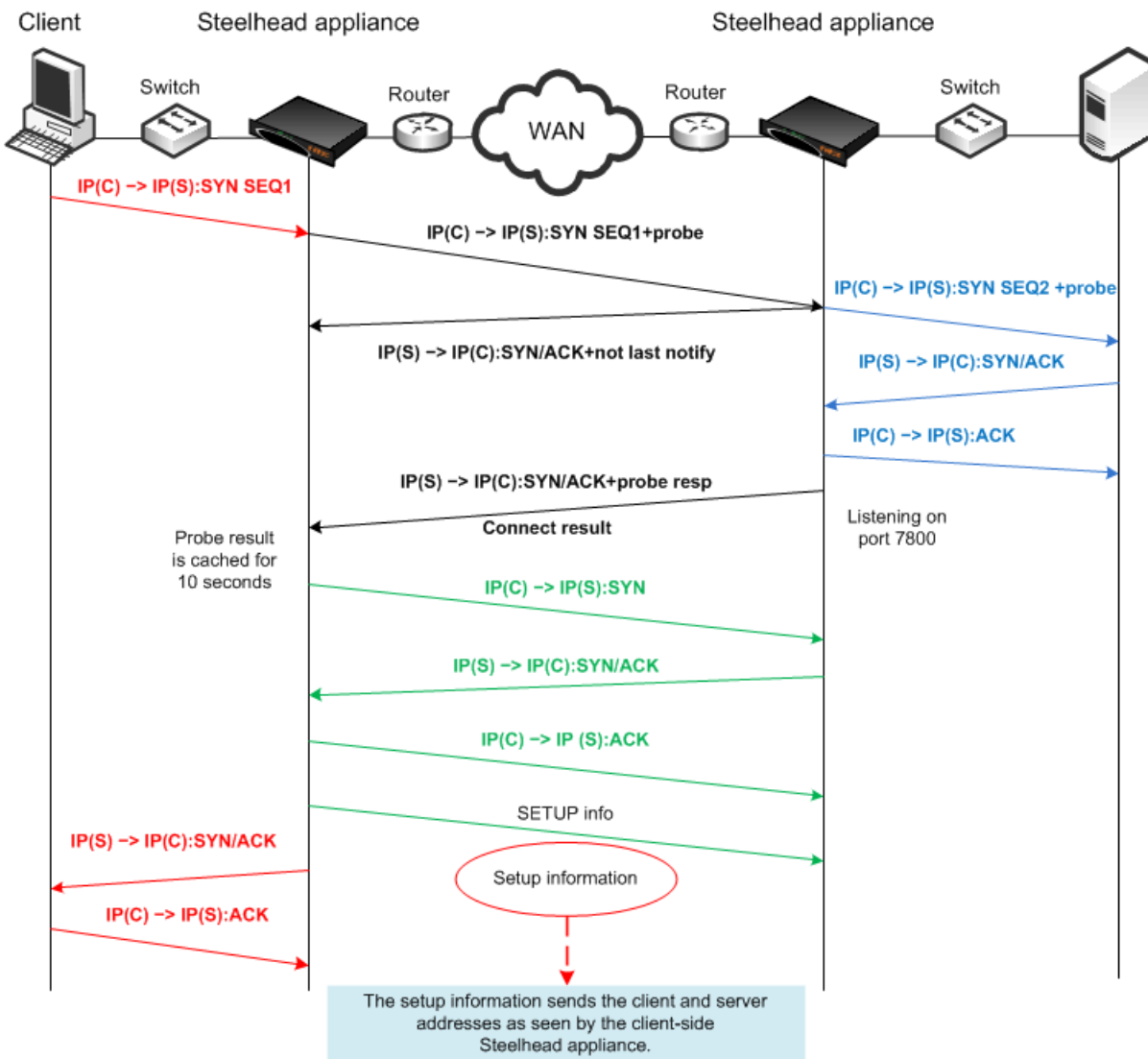
Network address translation (NAT) affects the addresses the Steelhead appliance uses in different ways, depending on which addressing mode is in use. This section provides several NAT deployment scenarios using various addressing modes:

- [“NAT Deployment Using Enhanced Auto-Discovery and Full Transparency” on page 61](#)
- [“NAT Deployment Using Fixed-Target Rules” on page 61](#)
- [“NAT Deployment Using Correct and Port Transparency Addressing Modes” on page 64](#)
- [“Client-Side Source NAT Using Enhanced Auto-Discovery and Full Transparency” on page 66](#)
- [“Failed Client-Side Source NAT Deployment Using Enhanced Auto-Discovery and Correct Addressing” on page 67](#)
- [“Dual NAT Deployment Using Enhanced Auto-Discovery and Full Transparency” on page 68](#)
- [“Failed Dual NAT Deployment Using Enhanced Auto-Discovery and Correct Addressing” on page 69](#)

NAT Deployment Using Enhanced Auto-Discovery and Full Transparency

Figure 3-8 shows full transparency addressing mode. The client and server IP addresses are preserved, and the presence of the full transparency TCP is a signal to the server-side Steelhead appliance that it can use the addresses arriving from the WAN. This ensures that any translation is preserved for optimized traffic.

Figure 3-8. Enhanced Auto-Discovery and Full Transparency Mode

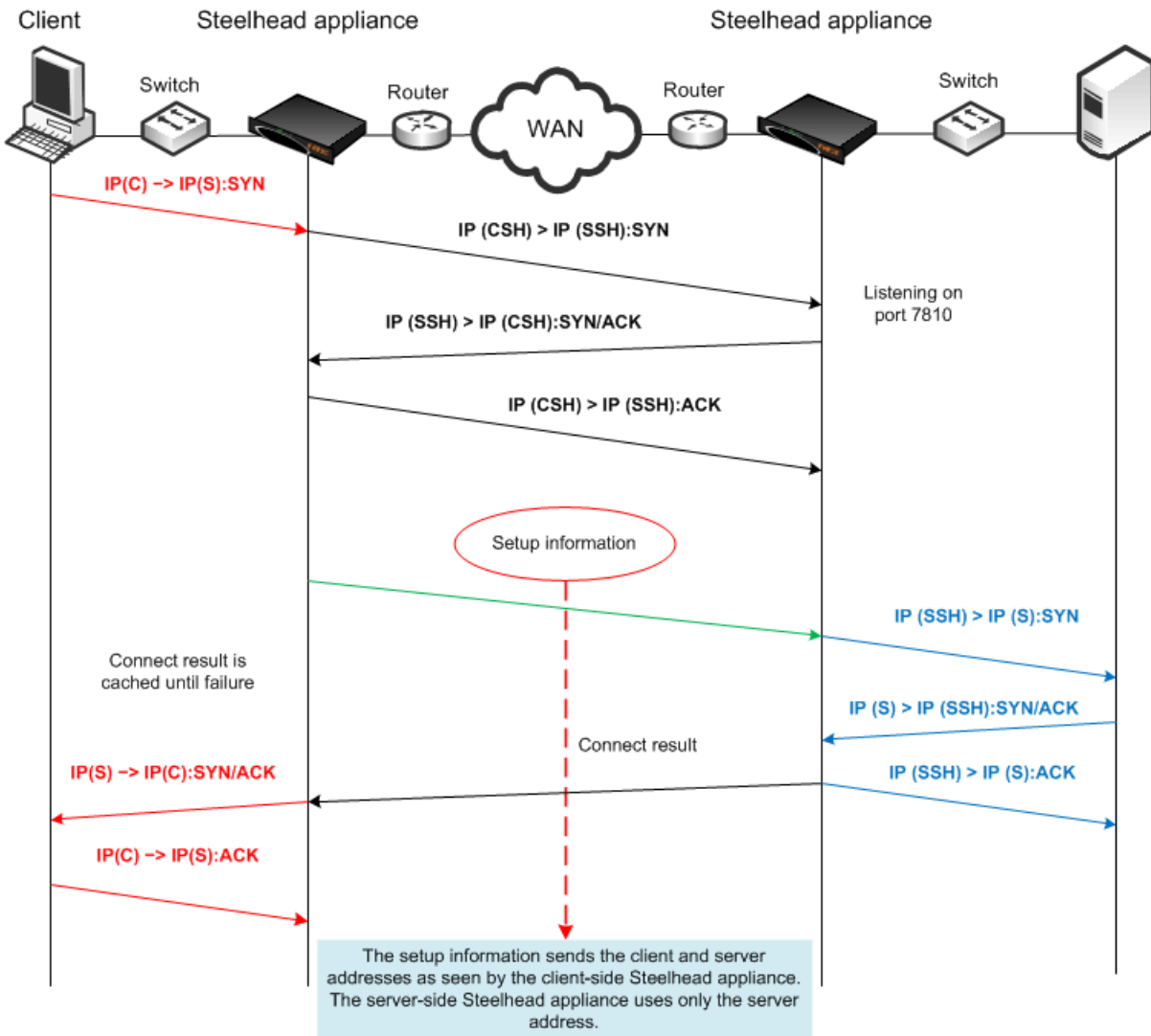


NAT Deployment Using Fixed-Target Rules

When you use a fixed-target rule to the primary IP address of a remote Steelhead appliance, the remote Steelhead appliance makes a connection to the destination IP address detected on the initiating Steelhead appliance. Figure 3-9 shows that the source address is the primary IP address of the remote Steelhead appliance.

When you use a fixed-target rule to the in-path IP address of a remote Steelhead appliance, the remote Steelhead appliance makes a connection to the destination IP address detected on the initiating Steelhead appliance. The remote Steelhead appliance also uses the same source address detected on the initiating Steelhead appliance.

Figure 3-9. Fixed-Target Rule to Primary IP Address



In the example, the out-of-path packet flow on incoming connection requests is very similar to an established in-path partnership. However, there is an important distinction in that the IP address of the server-side Steelhead appliance replaces the IP address of the client in communication between the server-side Steelhead appliance and the destination server. The traffic travels the following route:

1. The packet is created coming from the initiator client (C) IP address to the destination server (S) IP address.
2. The client-side Steelhead appliance sends a packet to port 7810 on the server-side Steelhead appliance, requesting to open a session.
3. The server-side Steelhead appliance acknowledges the connection request.

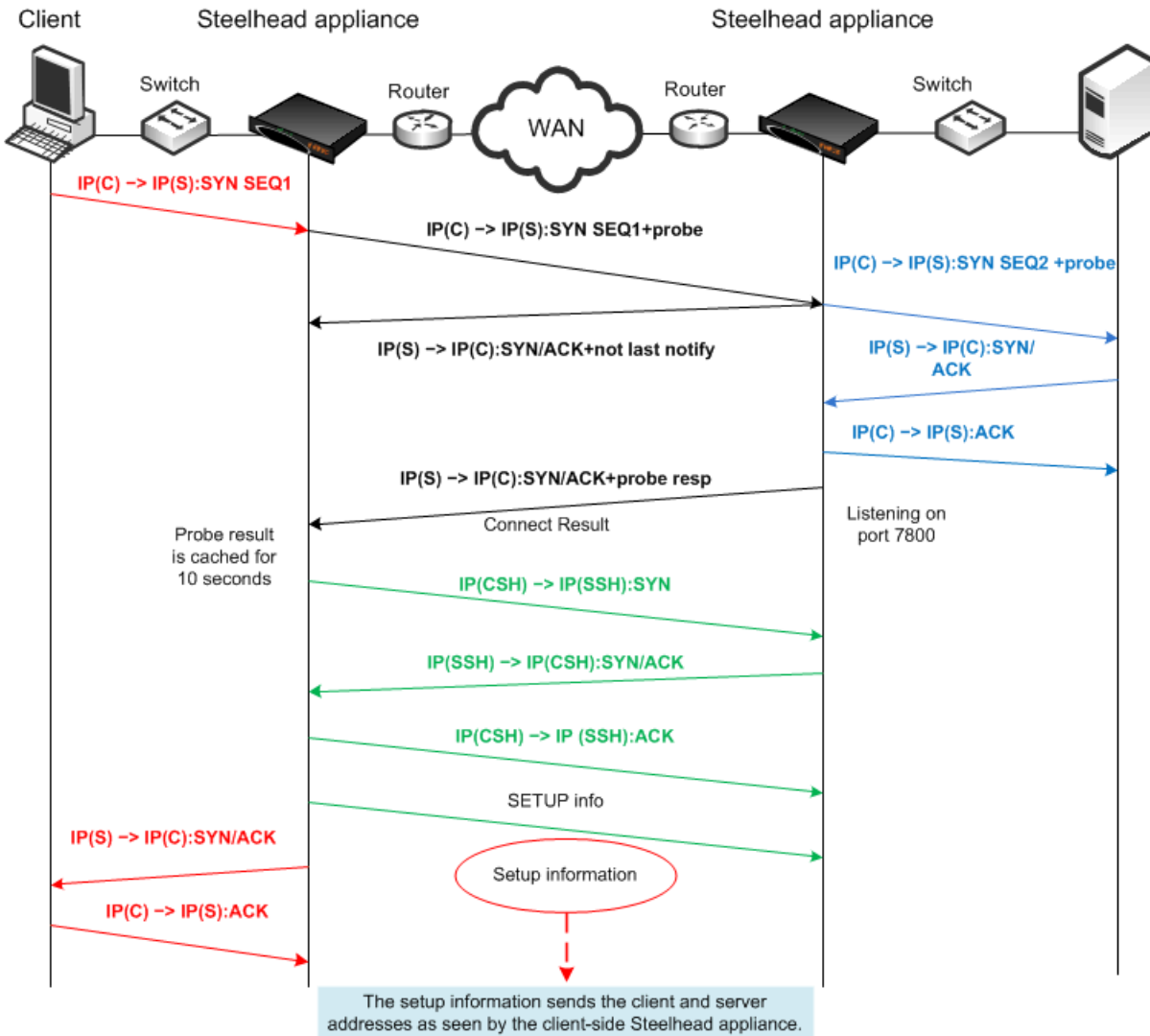
4. The client-side Steelhead appliance acknowledges the connection.
5. The client-side Steelhead appliance sends session setup information to the server-side Steelhead appliance.
6. The server-side Steelhead appliance forwards the original connection request to the destination server, replacing the client IP address with the server-side Steelhead appliance IP address.
7. The destination server acknowledges the connection request.
8. The server-side Steelhead appliance sends a packet acknowledgment to the destination server.
9. The server-side Steelhead appliance sends the connection acknowledgment to the client-side Steelhead appliance.
10. The client-side Steelhead appliance sends the acknowledgment packet to the requesting client.
11. The client sends an acknowledgment to the destination server.
12. The client-side Steelhead appliance discards the client acknowledgment.

Some applications and protocols require that the server initiate a new session or that they see the IP address of the requesting client. These applications and protocols do not function in this configuration. Consider using an in-path deployment, a WCCP deployment, or use rules on the Steelhead appliance to pass through this traffic.

NAT Deployment Using Correct and Port Transparency Addressing Modes

In both correct and port transparency addressing modes, whatever IP addresses are detected on the initiating Steelhead appliance (typically, the client-side Steelhead appliance) are used by the corresponding Steelhead appliance on the remote side, as [Figure 3-10](#) shows. This deployment can bypass any NAT that occurs in the WAN between the Steelhead appliances. To ensure that NAT is still used for the optimized traffic, you must configure the full transparency addressing mode for this traffic.

Figure 3-10. Enhanced Auto-Discovery in Correct and Port Transparency Modes



In this example, the TCP connection request travels the following route:

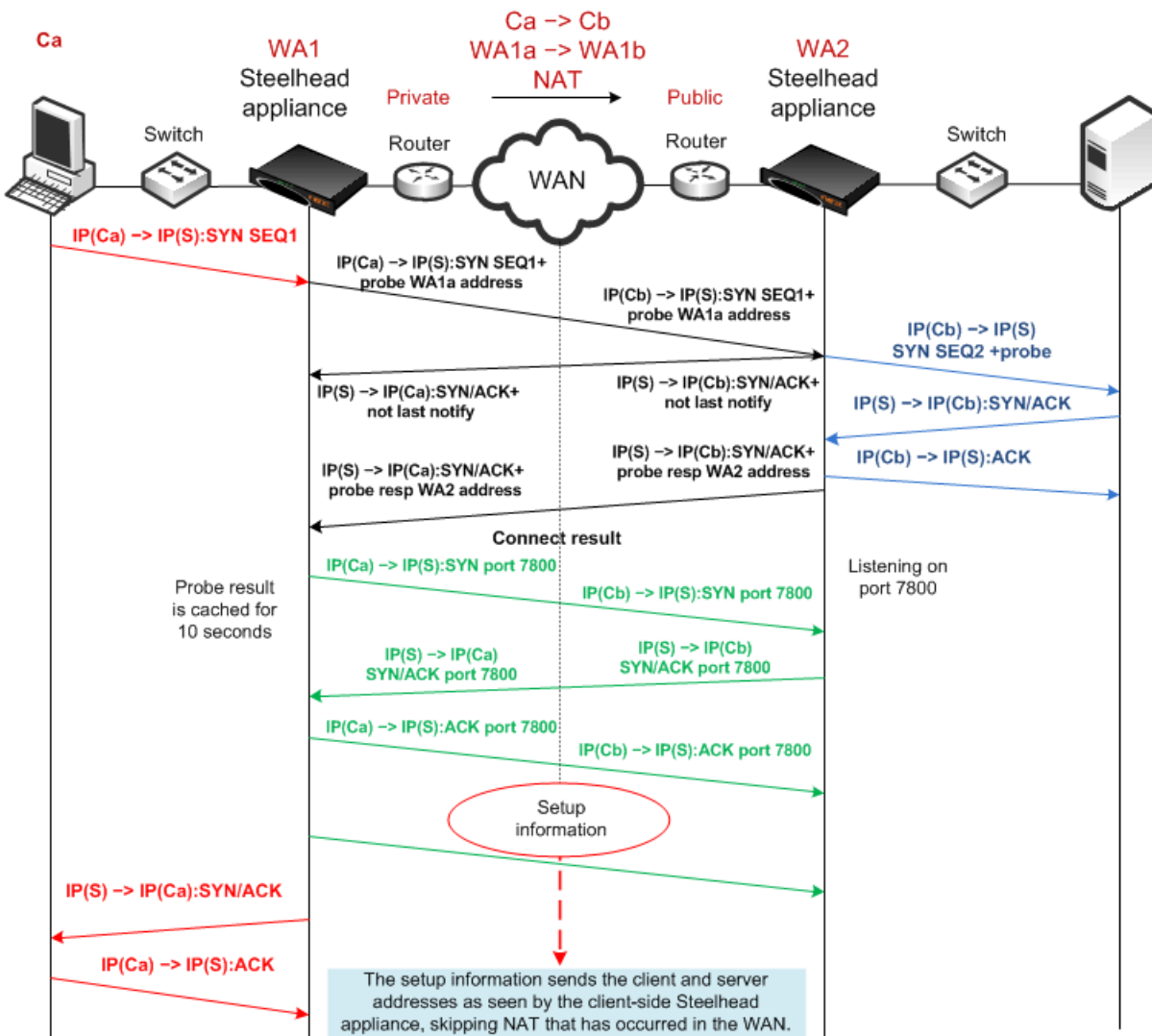
1. The packet is created coming from the initiator client (C) IP address to the destination server (S) IP address.
2. The client-side Steelhead appliance adds a probe to the TCP connection request.
3. The server-side Steelhead appliance sends *not last notify* immediately after receiving the probe.

4. The server-side Steelhead appliance forwards the original connection request to the destination server using client IP:port as source address, with a sequence number 2.
5. The destination server acknowledges the client connection request.
6. The server-side Steelhead appliance intercepts the return packet.
7. The server-side Steelhead appliance sends a packet acknowledgement to the destination server on behalf of the client.
8. The server-side Steelhead appliance acknowledges the client-side Steelhead appliance, and sends a notification with a probe response indicating that it is the last Steelhead appliance before the server.
9. The client-side Steelhead appliance sends a request to open port 7800 on the server-side Steelhead appliance.
10. The server-side Steelhead appliance acknowledges the connection request.
11. The client-side Steelhead appliance acknowledges the connection.
12. The client-side Steelhead appliance sends session setup information to the server-side Steelhead appliance.
13. The server-side Steelhead appliance connection sends an acknowledgment to the client-side Steelhead appliance.
14. The client-side Steelhead appliance sends the connection acknowledgment to the requesting client.
15. The client sends an acknowledgement to the destination server.
16. The client-side Steelhead appliance discards the client acknowledgment.

Client-Side Source NAT Using Enhanced Auto-Discovery and Full Transparency

Figure 3-11 shows a client-side source NAT deployment using enhanced auto-discovery and full address transparency. In this configuration, the presence of the full transparency TCP option 78 is a signal to the server-side Steelhead appliance that it can use the addresses arriving from the WAN. Because the server-side addresses are reachable from the client side, when the client-side Steelhead appliance makes its out-of-band connection to the server-side Steelhead appliance, the address it uses is valid and is properly translated across the WAN.

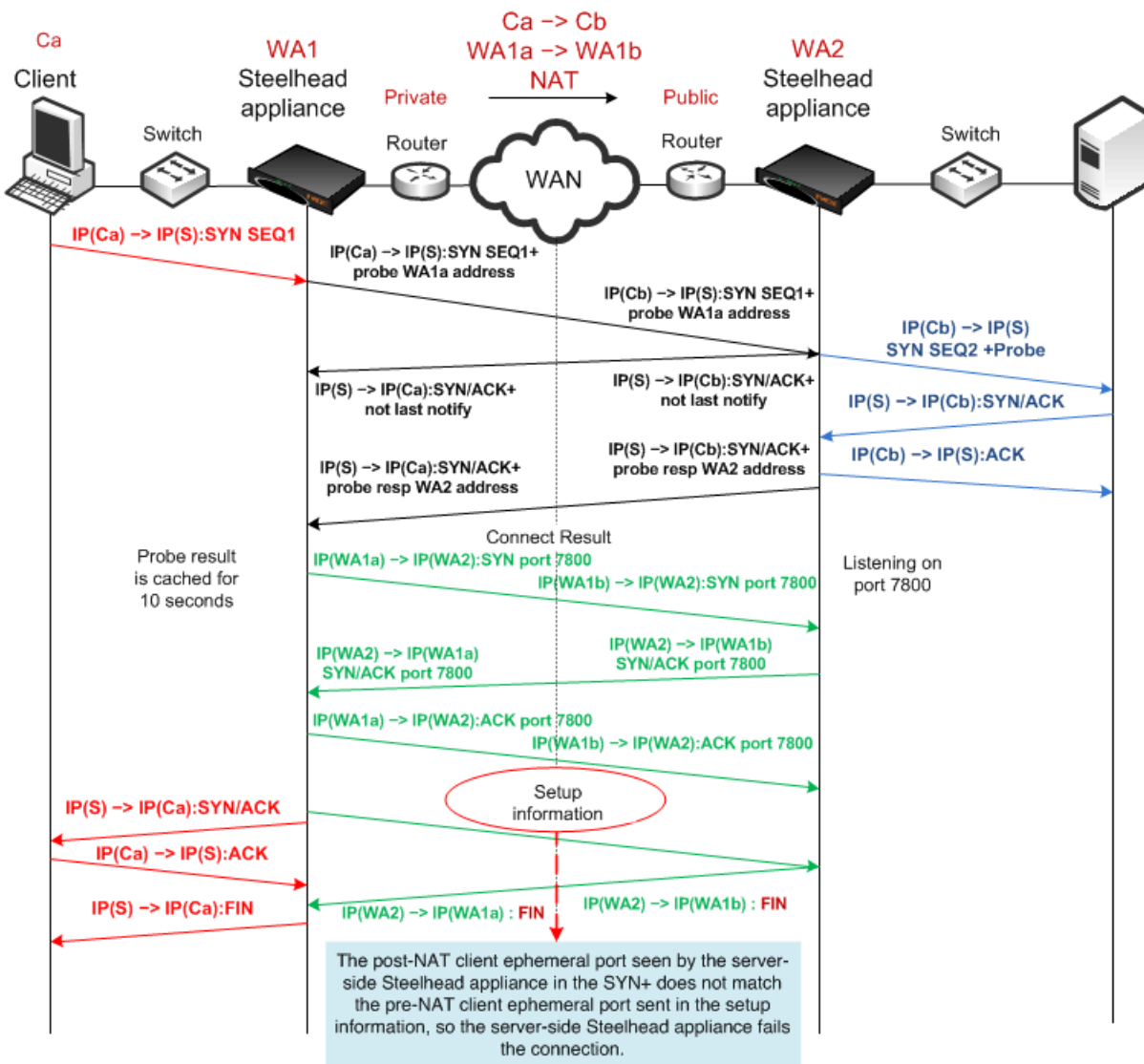
Figure 3-11. Full Transparency, Enhanced Auto-Discovery, and Client-Side Source NAT



Failed Client-Side Source NAT Deployment Using Enhanced Auto-Discovery and Correct Addressing

Figure 3-12 shows a deployment in which NAT is occurring at the client location. In this example, enhanced auto-discovery with correct addressing in does not work, because the server-side Steelhead cannot match the client ephemeral port to the ephemeral port used by the server-side Steelhead to connect to the server.

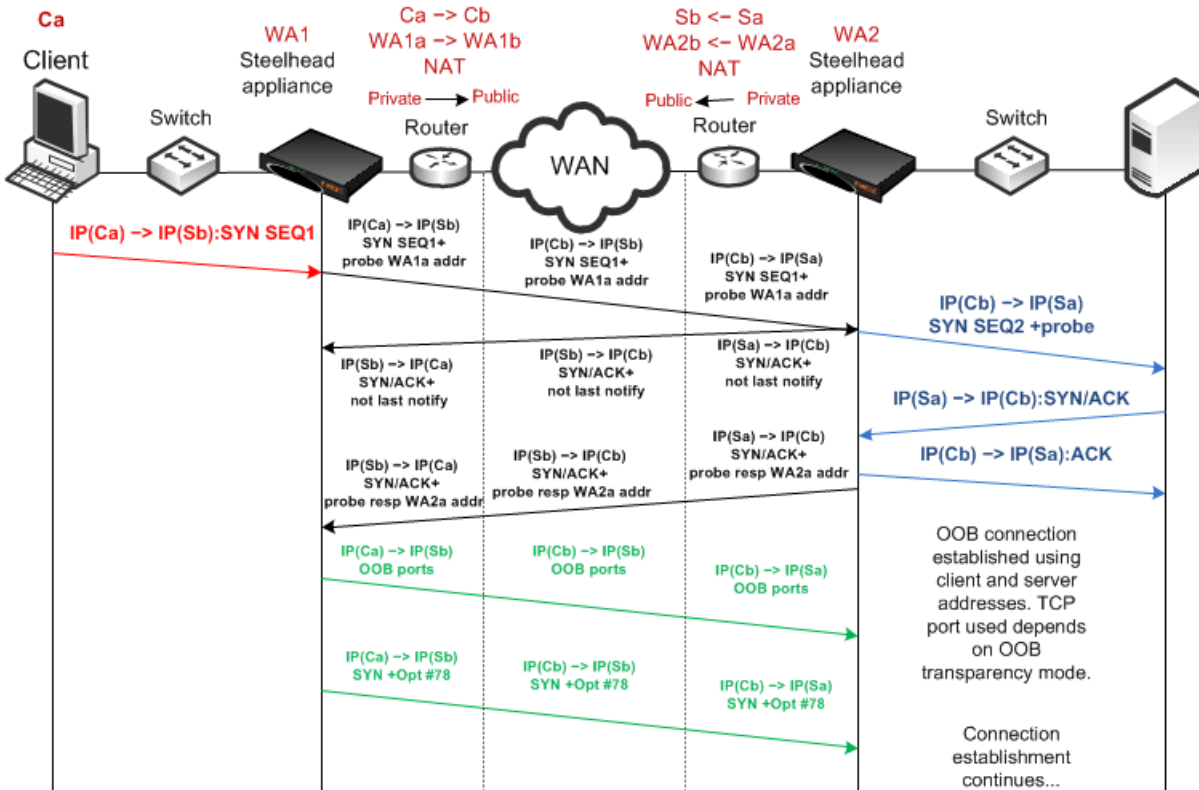
Figure 3-12. Enhanced Auto-Discovery, Correct Addressing, and Client-Side Source NAT



Dual NAT Deployment Using Enhanced Auto-Discovery and Full Transparency

Figure 3-13 shows NAT used at the client and the server. In this network, full transparency and some form of OOB transparency is required for successful connection establishment and optimization.

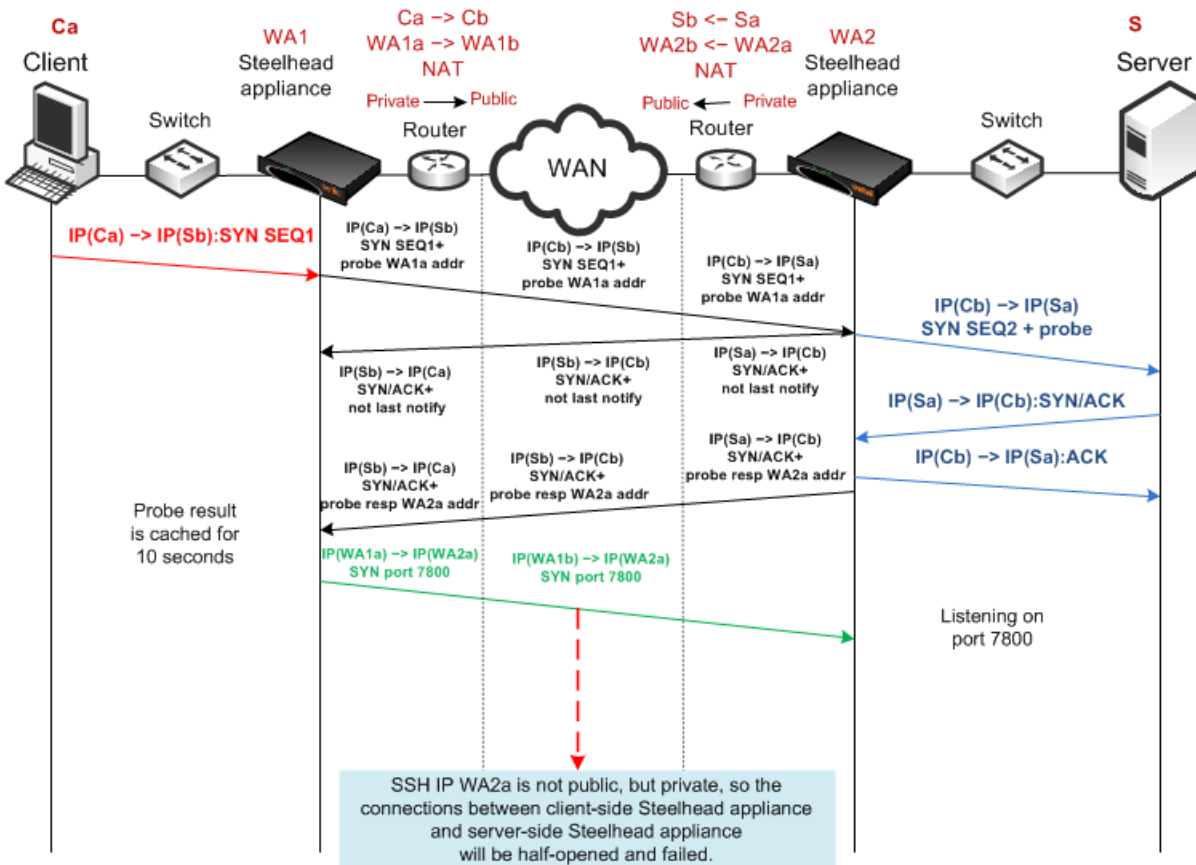
Figure 3-13. Enhanced Auto-Discovery, Full Transparency, OOB Transparency, and Dual NAT



Failed Dual NAT Deployment Using Enhanced Auto-Discovery and Correct Addressing

Figure 3-14 shows a deployment in which NAT is occurring at both the client and server locations. In this example, enhanced auto-discovery with correct addressing is unlikely to work, because the probe response from the server-side Steelhead appliance included the WA2a internal IP address for the server-side Steelhead appliance.

Figure 3-14. Enhanced Auto-Discovery, Correct Addressing, Dual NAT, Resulting in Half-Opened Connection



The Out-of-Band Connection

This section describes transparency options for the Out-of-Band (OOB) connection. This section includes the following topics:

- “Overview of OOB Connections and Addressing Modes” on page 70
- “Configuring OOB Connection Destination Transparency” on page 70
- “Configuring OOB Connection Full Transparency” on page 71

Overview of OOB Connections and Addressing Modes

A Steelhead appliance OOB connection is a TCP connection that Steelhead appliances establish with each other when they begin optimizing traffic to exchange capabilities and feature information, and to detect failures. A Steelhead appliance creates an OOB connection for each pair of local and remote in-path interfaces that are used when optimizing connections. OOB connections are created by the Steelhead appliance closest to the initiating side of the optimized connection.

The addresses and ports used by OOB connections depend on the addressing mode used for the first optimized connection between Steelhead appliances. If the addressing mode for the first connection is correct addressing or port transparency, the OOB connection uses correct addressing. If the first connection is full transparency, the default behavior is to make the OOB connection use correct addressing, but you can alter this behavior such that the connection uses a form of network transparency.

In some environments, it might be necessary to make OOB connections use some form of network transparency: for example, if network is unable to route between the in-path IP addresses or VLANs of Steelhead appliances that are optimizing traffic. Two options for OOB transparency exist:

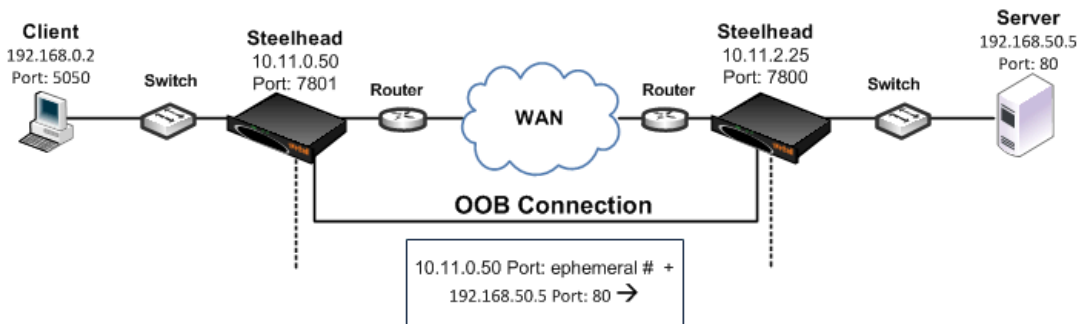
- Destination transparency
- Full transparency

The two options differ in which source IP address and TCP port are used. For details see [“Configuring OOB Connection Destination Transparency,”](#) next and [“Configuring OOB Connection Full Transparency”](#) on page 71.

Configuring OOB Connection Destination Transparency

Figure 3-15 shows TCP/IP packet headers when OOB connection destination transparency is enabled.

Figure 3-15. OOB Connection Destination Transparency



OOB connection destination transparency uses the following values in the TCP/IP packet headers in both directions across the WAN:

- Client-side Steelhead appliance IP address and an ephemeral port number chosen by the client-side Steelhead appliance + server IP address and port number.
- Steelhead appliances use the server IP address and port number from the first optimized connection.

Use OOB connection destination transparency if the client-side Steelhead appliance cannot establish the OOB connection to the server-side Steelhead appliance.

Note: You must first configure WAN visibility full address transparency for OOB connection destination transparency to function correctly.

To enable OOB connection destination transparency

- Connect to the CLI on the client-side Steelhead appliance and enter the following commands:

```
enable
configure terminal
in-path peering oobtransparency mode destination
write memory
```

Note: The changes take effect immediately. You must save your changes or they are lost upon reboot.

To disable OOB connection destination transparency

- Connect to the Riverbed CLI on the client-side Steelhead appliance and enter the following commands:

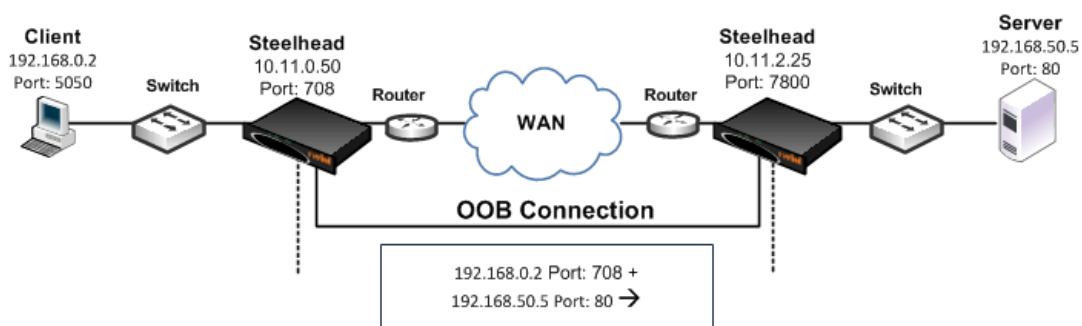
```
enable
configure terminal
in-path peering oobtransparency mode none
write memory
```

Note: The changes take effect immediately. You must save your changes or they are lost upon reboot.

Configuring OOB Connection Full Transparency

Figure 3-16 shows TCP/IP packet headers when OOB connection full transparency is enabled.

Figure 3-16. OOB Connection Full Transparency



OOB connection full transparency uses the following values in the TCP/IP packet headers in both directions across the WAN:

- Client IP address and client-side Steelhead appliance predetermined port number 708 + server IP address and port number.

Steelhead appliances use the client IP address, and the server IP address and port number from the first optimized connection.

If the client is already using port 708 to connect to the destination server, enter the following CLI command to change the client-side Steelhead appliance predetermined port number:

```
in-path peering oobtransparency port <port number>
```

OOB connection full transparency supports Steelhead appliances deployed on trunks. Because you can configure full address transparency so that optimized traffic remains on the original VLAN, you no longer need for a Steelhead appliance VLAN.

Use OOB connection full transparency if your network is unable to route between Steelhead appliance in-path IP addresses or in-path VLANs, or you do not want to see Steelhead appliance IP addresses used for the OOB connection.

You must first configure WAN visibility full address transparency for OOB connection full transparency to function correctly. For details, see [“Full Address Transparency” on page 53](#).

To enable OOB connection full transparency

- Connect to the CLI on the client-side Steelhead appliance and enter the following commands:

```
enable
configure terminal
in-path peering oobtransparency mode full
write memory
```

The changes take effect immediately. You must save your changes or they are lost upon reboot.

To disable OOB connection full transparency

- Connect to the CLI on the client-side Steelhead appliance and enter the following commands:

```
enable
configure terminal
in-path peering oobtransparency mode none
write memory
```

The changes take effect immediately. You must save your changes or they are lost upon reboot.

Configuring WAN Visibility Modes

This section describes how to configure WAN visibility modes using an example deployment and the RiOS CLI.

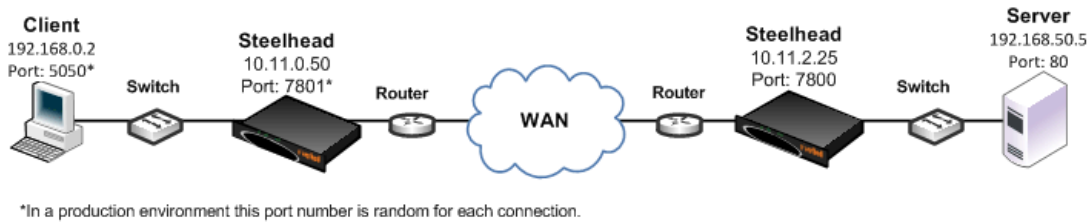
You configure WAN visibility modes by creating an in-path rule on the client-side Steelhead appliance (where the connection is initiated). By default, the rule is placed before the default in-path rule, and after the secure, interactive, and RBT-proto rules.

For transparent addressing to function correctly, both of the Steelhead appliances must have RiOS v5.0.x or later installed. If one Steelhead appliance does not support transparent addressing (that is, it has RiOS v4.1 or earlier installed), the Steelhead appliance attempting to optimize a connection in one of the transparent addressing modes automatically reverts to correct addressing mode, and optimization continues.

By default, Steelhead appliances use correct addressing (for all RiOS versions).

Figure 3-17 shows the IP addresses and ports used in the example deployments.

Figure 3-17. Example Deployment For Configuring WAN Visibility Modes



The following table summarizes the port transparency CLI commands.

Action	CLI Command
Enable port transparency for a specific server	in-path rule auto-discover wan-visibility port dstaddr 192.168.50.1/32 dstport 80
Enable full address transparency for a specific group of servers, and port transparency for servers not in the group	in-path rule auto-discover wan-visibility full dstaddr 192.168.0.0/24 in-path rule auto-discover wan-visibility port Important: In this example, the first in-path rule must precede the second in-path rule in the rule list. To specify the placement of a rule in the list, use the rulenum CLI command option. For details, see the <i>Riverbed Command-Line Interface Reference Manual</i> .
Disable port transparency	Delete the in-path rule that enables it. For details on deleting in-path rules, see the <i>Riverbed Command-Line Interface Reference Manual</i> .

The following table summarizes the full address transparency CLI commands.

Action	CLI Command
To enable full address transparency globally	in-path rule auto-discover wan-visibility full
To enable full address transparency for servers in a specific IP address range	in-path rule auto-discover wan-visibility full dstaddr 192.168.0.0/16
To enable full address transparency for a specific server	in-path rule auto-discover wan-visibility full dstaddr 192.168.50.1/32
To enable full address transparency for a specific group of servers, and port transparency for servers not in the group	in-path rule auto-discover wan-visibility full dstaddr 192.168.0.0/24 in-path rule auto-discover wan-visibility port Important: In this example, the first in-path rule must precede the second in-path rule in the rule list. To specify the placement of a rule in the list, use the rulenum CLI command option. For details, see the <i>Riverbed Command-Line Interface Reference Manual</i> .
To disable full address transparency	Delete the in-path rule that enables it. For details on deleting in-path rules, see the <i>Riverbed Command-Line Interface Reference Manual</i> .

CHAPTER 4 QoS Configuration and Integration

This chapter describes how to integrate Steelhead appliances into existing Quality of Service (QoS) architectures, and how to configure Riverbed QoS in RiOS. Additionally, this chapter describes how to use MX-TCP.

This chapter includes the following sections:

- “Overview of Riverbed QoS” on page 76
- “Integrating Steelhead Appliances into Existing QoS Architectures” on page 78
- “Application Flow Engine” on page 83
- “Basic Outbound QoS” on page 84
- “Advanced Outbound QoS” on page 88
- “Inbound QoS” on page 98
- “Guidelines for the Maximum Number of QoS Classes, Sites, and Rules” on page 100
- “LAN Bypass” on page 103
- “QoS for IPv6” on page 103
- “QoS in Virtual In-Path and Out-of-Path Deployments” on page 104
- “QoS in Multiple Steelhead Appliance Deployments” on page 104
- “QoS and Multiple WAN Interfaces” on page 105
- “QoS Enforcement Best Practices” on page 105
- “Migrating Between Basic and Advanced Outbound QoS Modes” on page 107
- “Upgrading to RiOS v6.5 or Later” on page 108

Important: If you are using a release previous to RiOS v8.5.2, some of the features described in the chapter might not be applicable. For details on QoS before RiOS v8.5.2, see earlier versions of the *Steelhead Appliance Deployment Guide* and *Riverbed Deployment Guide* on the Riverbed Support site at <https://support.riverbed.com>.

For more details on Riverbed QoS, see the *Steelhead Appliance Management Console User’s Guide*.

For more configuration examples, see “QoS Configuration Examples” on page 109.

Overview of Riverbed QoS

This section introduces QoS and Riverbed QoS. This section includes the following topics:

- [“Introduction to Riverbed QoS” on page 76](#)
- [“Enforcing QoS Policies using Riverbed QoS” on page 78](#)

Introduction to Riverbed QoS

Riverbed QoS is complementary to RiOS WAN optimization. Whereas SDR, transport streamlining, and application streamlining techniques work best with bandwidth-hungry or thick applications such as email, file-sharing, and backup, QoS helps best with latency-sensitive or thin applications like VoIP and interactive applications. QoS depends on accurate classification of traffic, bandwidth reservation, and proper traffic priorities.

Another way to look at the two is that WAN optimization speeds up some traffic by reducing its bandwidth needs and accelerating it, and QoS slows down some traffic to guarantee latency and bandwidth to other traffic. A combination of both techniques is ideal.

Because the Steelhead appliance acts as a TCP proxy, the appliance already works with traffic flows. Riverbed QoS is an extensive flow-based QoS system, which can queue traffic on a per-flow basis and uses standard TCP mechanics for traffic shaping to avoid packet loss on a congested link. Steelhead appliances support inbound and outbound QoS.

The major functionalities of QoS are this:

- **Classification** - Identifies and groups traffic. Riverbed QoS can do this based on TCP/UDP-header information, VLAN ID, or by using the Riverbed Application Flow Engine (AFE). Identified traffic is grouped into classes and/or QoS marked (DSCP or TOS).
For more information about AFE, see [“Application Flow Engine” on page 83](#).
- **Policing** - Defines the action against the classified traffic. Riverbed QoS can define a minimum and maximum bandwidth per class, the priority of a class relative to other classes, and a weight for the usage of excess bandwidth (unused bandwidth, which is allocated to other classes).
- **Enforcement** - Where the action takes place. Enforcement is performed using the Steelhead appliance QoS scheduler, which is based on the Hierarchical Fair Service Curve (HFSC) algorithm.

You can perform policing and enforcement on a downstream networking device when classification and QoS marking are performed on the Steelhead appliance. This is useful if the Steelhead appliance must integrate with an existing QoS implementation.

Riverbed QoS is flow based. For TCP, the Steelhead appliance must detect the three-way handshake or it cannot classify a traffic flow. If a traffic flow is not classified, it falls into the default class.

After a traffic flow is classified, it is registered and cannot be reclassified. If a flow changes to a different application and is not reset by the application, the classification stays the same as before the change. Steelhead appliances set up 100-packet deep packet buffers per configured class, regardless of the packet size. You can adjust the depths of the packet buffers using the CLI.

Riverbed QoS takes effect as soon as traffic congestion occurs on a link. Congestion occurs when multiple flows are sending data at the same time and the packets of the flows are not forwarded immediately, forming a queue.

There are two types of congestion: long term, which lasts a second or longer, and short term, which occurs for less than a second. Both types of congestion signal that one or more applications are slowed down because of other traffic.

You can best manage long-term congestion by shaping traffic: reserving bandwidth for the more important traffic. This is the most well-known implementation for QoS.

Short-term congestion is what can cause applications to *hang* for a second, or reduce the quality of a VoIP conversation. You can manage short-term congestion by prioritizing of traffic—packets of latency sensitive traffic are moved ahead of the queue.

A well-designed QoS environment uses both prioritization and traffic shaping to guarantee bandwidth and latency for applications.

Many QoS implementations use some form of packet fair queuing (PFQ), such as weighted fair queuing (WFQ) or class-based weighted fair queuing (CBWFQ). As long as high-bandwidth traffic requires a high priority (or vice-versa), PFQ systems perform adequately. However, problems arise for PFQ systems when the traffic mix includes high-priority, low-bandwidth traffic (such as VoIP), or high-bandwidth traffic that does not require a high priority (such as e-mail), particularly when both of these traffic types occur together.

Additional features such as low-latency queuing (LLQ) attempt to address these concerns by introducing a separate system of strict priority queuing that is used for high-priority traffic. However, LLQ is not a principled way of handling bandwidth and latency trade-offs. LLQ is a separate queuing mechanism meant as a work-around for PFQ limitations.

The Riverbed QoS system is based on a patented version of HFSC. HFSC allows bandwidth allocation for multiple applications of varying latency sensitivity. HFSC explicitly considers delay and bandwidth at the same time. Latency is described in six priority levels (real-time, interactive, business critical, normal, low, and best-effort) that you assign to classes.

If you assign a priority to a class, the class can tolerate X delay, in which X is the priority setting. At the same time, bandwidth guarantees are respected.

This enables Riverbed to deliver low latency to traffic without wasting bandwidth, and deliver high bandwidth to delay-insensitive traffic without disrupting delay-sensitive traffic.

The Riverbed QoS system achieves the benefits of LLQ without the complexity and potential configuration errors of separate, parallel queuing mechanisms.

For example, you can enforce a mix of high-priority, low-bandwidth traffic patterns (SSH, Telnet, Citrix, RDP, CRM systems, and so on) with lower-priority, high-bandwidth traffic (FTP, backup, replication, and so on). This enables you to protect delay-sensitive traffic such as VoIP, alongside other delay-sensitive traffic such as video conferencing, RDP, and Citrix. You can do this without having to reserve large amounts of bandwidth for the traffic classes.

You can apply Riverbed QoS to both pass-through and optimized traffic, and it does not require the optimization service. QoS classification occurs during connection setup for optimized traffic - before optimization and compression. QoS shaping and enforcement occurs after optimization and compression. Pass-through traffic has the QoS shaping and enforcement applied appropriately. However, with the introduction of the Steelhead CX and Steelhead EX, there are platform-specific limits defined for the following QoS settings for outbound QoS:

- Maximum configurable root bandwidth
- Maximum number of classes
- Maximum number of rules
- Maximum number of sites

There are no platform-specific limits for Inbound QoS.

For more details on limits, see [“Guidelines for the Maximum Number of QoS Classes, Sites, and Rules” on page 100](#).

You can perform differentiated services code point (DSCP) marking and QoS enforcement on the same traffic. First mark the traffic, and then perform QoS qualification and management on the post-marked traffic.

For details on marking traffic, see [“QoS Marking” on page 80](#).

Enforcing QoS Policies using Riverbed QoS

You apply Riverbed QoS policies in the Configure > Networking > Outbound QoS (Basic), Outbound QoS (Advanced), and Inbound QoS pages. The following sections of this chapter describe the specifics.

The main components of the Riverbed QoS enforcement system are QoS classes and QoS rules. A QoS class represents an aggregation of traffic that is all treated the same way by the QoS scheduler. QoS rules determine membership of traffic in a particular QoS class, and are based on the following parameters: IP addresses, protocols, ports, DSCP, traffic type (optimized and pass-through), VLAN tags, applications, and services. The QoS scheduler uses the constraints and parameters configured on the QoS classes, such as minimum bandwidth guarantee and latency priority, to determine in what order packets are transmitted from the Steelhead appliance.

Riverbed QoS supports shaping and marking for multicast and broadcast traffic. To classify this type of traffic, you must configure IP header-based rules, because the Application Flow Engine (AFE) does not support multicast and broadcast traffic.

For details on the AFE, see [“Application Flow Engine” on page 83](#).

Integrating Steelhead Appliances into Existing QoS Architectures

This section describes the integration of Steelhead appliances into existing QoS architectures. This section includes the following topics:

- [“WAN-Side Traffic Characteristics and QoS” on page 79](#)
- [“QoS Integration Techniques” on page 79](#)
- [“QoS Marking” on page 80](#)

When you integrate Steelhead appliances into your QoS architecture, you can:

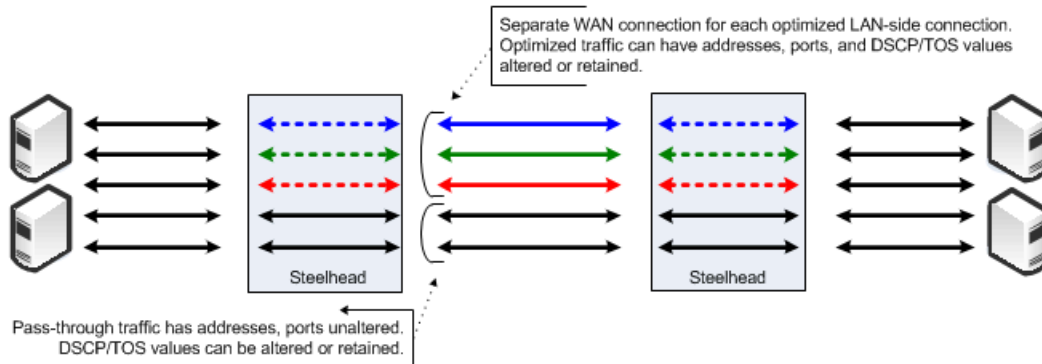
- retain the original DSCP or IP precedence values.
- choose the DSCP or IP precedence values.
- retain the original destination TCP port.
- choose the destination TCP port.
- retain all of the original IP addresses and TCP ports.

You do not have to use all of the Steelhead appliance functions on your optimized connections. You can selectively apply functions to different optimized traffic, based on attributes such as IP addresses, TCP ports, DSCP, VLAN tags, and payload.

WAN-Side Traffic Characteristics and QoS

When you integrate Steelhead appliances into an existing QoS architecture, it is helpful to understand how optimized and pass-through traffic appear to the WAN, or any WAN-side infrastructure. [Figure 4-1](#) shows how traffic appears on the WAN when Steelhead appliances are present.

Figure 4-1. How Traffic Appears to the WAN when Steelhead Appliances Are Present



When Steelhead appliances are present in a network:

- the optimized data for each LAN-side connection is carried on a unique WAN-side TCP connection.
- the IP addresses, TCP ports, and DSCP or IP precedence values of the WAN connections are determined by the Steelhead appliance WAN visibility mode, and the QoS marking settings configured for the connection.
- the amount of bandwidth and delay assigned to traffic when you enable Riverbed QoS enforcement is determined by the Riverbed QoS enforcement configuration. This applies to both pass-through and optimized traffic. However, this configuration is separate from configuring Steelhead appliance WAN visibility modes.

For details on WAN visibility modes, see [“Overview of WAN Visibility” on page 49](#).

QoS Integration Techniques

This section provides examples of different QoS integration techniques, depending on the environment:

- [“QoS Policy Differentiating Voice Versus Non-Voice Traffic” on page 80](#)
- [“Steelhead Appliance Honoring Pre-Marked LAN-Side Traffic” on page 80](#)
- [“Steelhead Appliance Re-Marking the LAN-Side Traffic” on page 80](#)
- [“Steelhead Appliance Enforcing the QoS Policy” on page 80](#)

These examples assume that the post-integration goal is to treat optimized and non-optimized traffic in the same manner with respect to QoS policies; some administrators might want to allocate different network resources to optimized traffic.

For details on QoS marking, see [“QoS Marking” on page 80](#).

In networks where both classification or marking and enforcement are performed on traffic after it passes through the Steelhead appliance, you have the following configuration options:

- In a network where classification and enforcement is based only on TCP ports, you can use port mapping, or the port transparency WAN visibility mode.

For details on port transparency, see [“Port Transparency” on page 52](#).

- In a network where classification and enforcement is based on IP addresses, you can use the full address transparency WAN visibility mode.

For details on full address transparency, see [“Full Address Transparency” on page 53](#).

QoS Policy Differentiating Voice Versus Non-Voice Traffic

In some networks, QoS policies do not differentiate traffic that is optimized by the Steelhead appliance. For example, because VoIP traffic is passed through the Steelhead appliance, a QoS policy that gives priority to only VoIP traffic, without differentiating between non-VoIP traffic, is unaffected by the introduction of Steelhead appliances. In these networks, you do not need to make QoS configuration changes to maintain the existing policy, because the configuration treats all non-VoIP traffic identically, regardless of whether it is optimized by the Steelhead appliance.

Steelhead Appliance Honoring Pre-Marked LAN-Side Traffic

Another example of a network that might not require QoS configuration changes to integrate Steelhead appliances is where traffic is marked with DSCP or TOS values before reaching the Steelhead appliance, and enforcement is made after reaching the Steelhead appliances based only on DSCP or TOS. The default Steelhead appliance settings reflect the DSCP/TOS values from the LAN-side to the WAN-side of an optimized connection.

For example, you configure QoS by marking the DSCP values at the source or on LAN-side switches, and enforcement is performed on WAN routers, the WAN routers detect the same DSCP values for all classes of traffic, optimized or not.

Steelhead Appliance Re-Marking the LAN-Side Traffic

You can mark or re-mark a packet with a different DSCP or TOS value as it enters the Steelhead appliance. This re-marking process is sometimes necessary because it is possible for an end host to set an inappropriate DSCP value for traffic that might otherwise receive a lower priority. In this scenario, the QoS enforcement remains the responsibility of the WAN router.

Steelhead Appliance Enforcing the QoS Policy

Instead of enforcing the QoS policy on the WAN router, you can configure the Steelhead appliance to enforce the QoS policy instead. However, Riverbed recommends that you also maintain a QoS policy on the WAN router to ensure that the traffic quality is guaranteed in the event that the Steelhead appliance becomes unavailable.

QoS Marking

This section describes how to use Steelhead appliance QoS marking when integrating Steelhead appliances into an existing QoS architecture. This section includes the following topics:

- [“QoS Marking for Optimized Traffic \(Global DSCP\)” on page 81](#)
- [“QoS Marking Default Setting” on page 81](#)
- [“QoS Marking in RiOS v7.0 and Later” on page 82](#)
- [“Upgrading QoS Marking to RiOS v7.0 from an Earlier RiOS Version” on page 82](#)

Steelhead appliances can retain or alter the DSCP or IP TOS value of both pass-through traffic and optimized traffic. In basic outbound QoS mode, you can alter the DSCP or IP TOS value in the service policies and in the application rules. In advanced outbound QoS mode, you can change the DSCP or IP TOS value in the QoS class or rules configuration.

For example QoS marking configurations, see [“Configuring QoS Marking on Steelhead Appliances” on page 121](#).

Note: In RiOS v7.0, the DSCP or IP TOS value definition changed significantly from earlier versions of RiOS. If you are running an earlier version of the RiOS, see an earlier version of the deployment guide for instructions on how to configure the DSCP or IP TOS value.

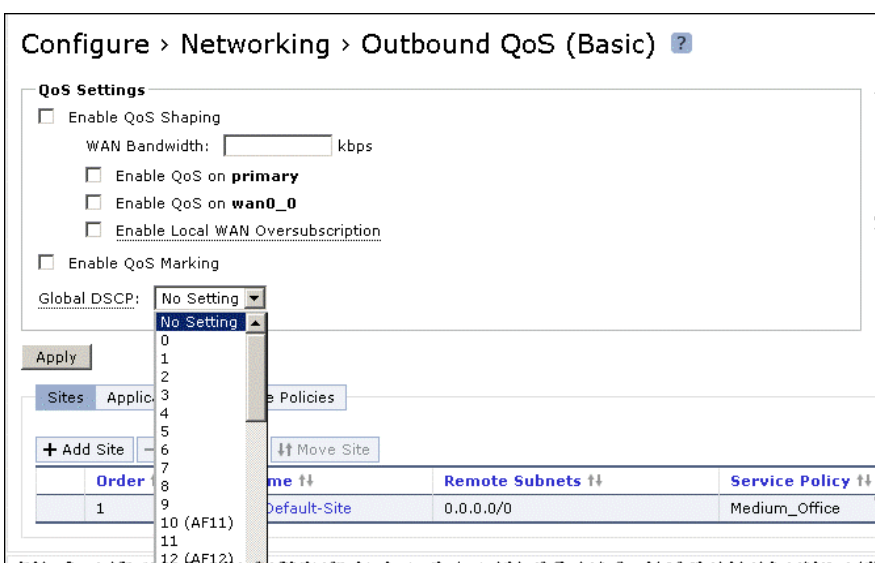
QoS Marking for Optimized Traffic (Global DSCP)

By default, the setup of optimized connections and the out-of-band control connections are not marked with a DSCP value. Existing traffic marked with a DSCP value is classified into the default class.

If your existing network provides multiple classes of service based on DSCP values, and you are integrating a Steelhead appliance into your environment, you can use the Global DSCP feature to prevent dropped packets and other undesired effects.

Global DSCP is available on basic and advanced outbound QoS.

Figure 4-2. Global DSCP



You can enable QoS marking without enabling QoS shaping.

QoS reporting does not show any output if QoS marking is enabled without enabling QoS shaping. To get reporting for marked traffic, you must enable QoS shaping.

QoS Marking Default Setting

By default, Steelhead appliances reflect the DSCP or IP TOS value found on pass-through traffic and optimized connections. The default value for DSCP or IP TOS are set to:

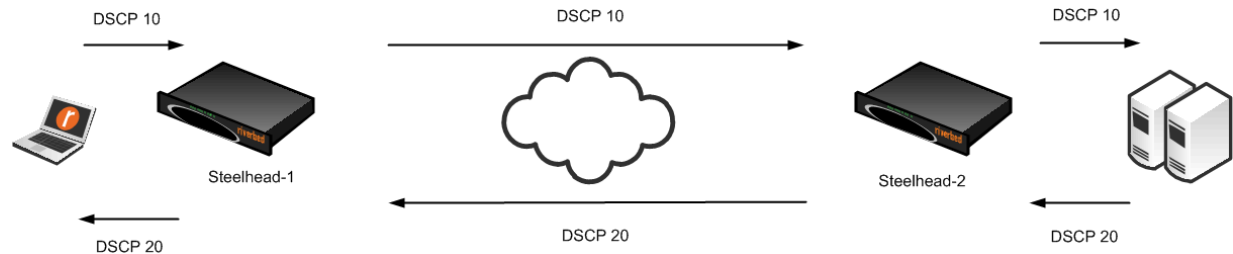
- *reflect* in the service class (basic outbound QoS) or in the QoS class (advanced outbound QoS).

- *inherit from service class* in the application rules (basic outbound QoS) or in the rules configuration (advanced outbound QoS).

By default, the DSCP or IP TOS value on pass-through or optimized traffic is unchanged when it passes through the Steelhead appliance.

Figure 4-3 shows reflected DSCP or IP TOS value detected on a network.

Figure 4-3. Reflected DSCP or IP TOS Value



QoS Marking in RiOS v7.0 and Later

Prior to RiOS v7.0, the DSCP or IP TOS value on a server-side Steelhead appliance was determined by the DSCP or IP TOS value of the client-side Steelhead appliance. In RiOS v7.0 or later, the server-side and client-side Steelhead appliances use the local QoS marking settings for DSCP or IP TOS marking.

Note: If the client-side Steelhead appliance is running RiOS v7.0 and the server-side Steelhead appliance is running an earlier RiOS version, then optimized traffic on the server-side Steelhead appliance is not marked with a DSCP or IP TOS value.

With RiOS v7.0 or later:

- you can use a DSCP or IP TOS value to classify traffic. You can use classification rules on an existing DSCP or IP TOS value instead of specifying individual header or Application Flow rules. This simplifies the configuration, because traffic is marked before it is classified.
- it is no longer possible to use QoS marking for traffic exiting the LAN interface. The LAN interface reflects the QoS marking value it receives from the WAN interface, for both optimized and unoptimized traffic.

QoS marking behavior is the same for in-path, virtual in-path, or out-of-path deployments. The one exception is that you can mark unoptimized traffic in a virtual in-path deployment such as DSCP or IP TOS with rules configured for LAN traffic.

Upgrading QoS Marking to RiOS v7.0 from an Earlier RiOS Version

Your existing QoS marking settings change when you upgrade to RiOS v7.0 because the DSCP or IP TOS value definitions have changed significantly in RiOS v7.0. If you have only configured QoS marking prior to the upgrade, the QoS configuration migrates to advanced outbound QoS mode. The DSCP or IP TOS value of the default class is set to *DSCP reflect*. Your original QoS marking settings are added as QoS rules to the default QoS rule set, and point to the default service class, which is *Default-Site\$\$Best-Effort*.

If you have configured QoS marking and traffic shaping prior to the upgrade, the QoS marking settings are set to the default values (for details, see [“QoS Marking Default Setting” on page 81](#)).

RiOS v7.0 uses a new format for the QoS shaping settings that do not allow you to adopt the QoS marking rules.

In previous RiOS versions, QoS marking rules are written in a human-readable format to a log file. You can reconfigure these rules for RiOS v7.0. You can retrieve the log file from the Steelhead appliance by using SCP: for example,

```
SCP <Steelhead_IP-address:/var/log/qos_old_mark_cfg_log.txt ./)
```

When you downgrade a Steelhead appliance to an earlier RiOS version, all changes to the QoS configuration are lost. The Steelhead appliance uses the most recent valid settings used with the RiOS version it is downgraded to.

Application Flow Engine

This section includes the following topics:

- [“Overview of Application Flow Engine” on page 83](#)
- [“AFE and Microsoft Lync” on page 84](#)

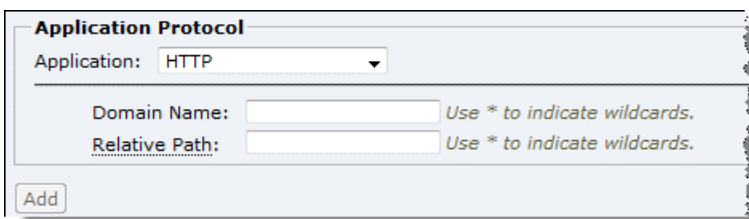
Overview of Application Flow Engine

With Application Flow Engine (AFE), QoS can identify applications accurately, and differentiate applications that use the same port on the same server. For example, Sharepoint and Microsoft BITS can use port 80 on the same server. After an application is identified, you can place it into different classes for QoS enforcement. The AFE is similar to deep packet inspection (DPI) because it identifies applications based on patterns. This approach allows you to more accurately identify modern applications than signature-based DPI methods by being aware of the more complex ways they communicate and the dependencies between multiple flows.

To view a completed global application list, see the *Steelhead Appliance Management Console User's Guide*.

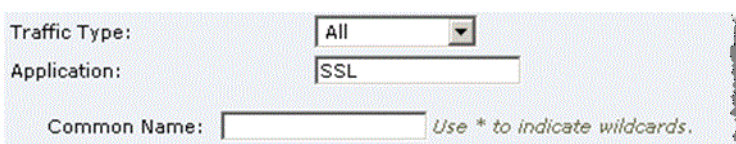
In addition to AFE supporting many well-known applications, you can add rules to identify custom applications. For example, you can identify a new HTTP application based on specific domain name or relative path.

Figure 4-4. Custom Application



With RiOS v8.6 or later, you can use the AFE to classify unoptimized SSL traffic based on the TLS/SSL server common name in the client certificate. To do this, add an application, choose SSL, and enter the common name of the client certificate in the respective field. To make the configuration easier, you can use wildcards in the name.

Figure 4-5. SSL Traffic



You cannot classify SSL optimized traffic using the common name control. For optimized or decrypted SSL traffic, the AFE uses the same techniques as non-encrypted traffic to classify the traffic. You can create a QoS HTTP rule to match the domain and server name for optimized or decrypted traffic.

For an example configuration, see [“Configuring QoS for SSL Common Name Matching” on page 142](#).

You can use AFE identification for pass-through and optimized traffic. The AFE is available in both the basic and advanced outbound QoS modes.

For details on QoS rules using the AFE, see [“QoS Rules” on page 94](#).

AFE and Microsoft Lync

RiOS v8.6 enhances support for Microsoft Lync. Lync is a multiple-feature communication suite that carries traffic over an extensive selection of protocols. The AFE classification of Lync traffic covers the majority of traffic generated between Lync clients and Lync servers.

This table lists the types of traffic Lync generates and the classification the AFE provides for them:

Workload	Classified As
Client login	LYNC, LYNCCTRL
Chat message	LYNC, LYNCCTRL
File transfer	LYNC, LYNCSHRE
Group voice chat	LYNC, LYNCMDIA
Video call	LYNC, LYNCMDIA
Application screen sharing	LYNC, LYNCSHRE
Desktop sharing	LYNC, LYNCSHRE
Voice call	LYNC, LYNCMDIA
Presentation sharing	SSL, SIP
White-board session	SSL, SIP

Note: A Lync server uses the default SIP port of TCP 5061. You can use this information to build a custom rule to classify Lync SIP traffic.

Basic Outbound QoS

The basic outbound QoS mode simplifies QoS task list, reduces the number of steps necessary to configure QoS. This section includes the following topics:

- [“Configuring Basic Outbound QoS” on page 85](#)
- [“Basic Outbound QoS Mode Restrictions” on page 86](#)
- [“WAN Oversubscription” on page 87](#)

Although the QoS functionality in RiOS versions earlier than v6.5 offer a greater degree of flexibility, the configuration can be complex. QoS configuration in RiOS v6.5 or later is easier to configure because of basic outbound QoS mode, which includes the *global application list*. The global application list is a list of default applications.

For details on adding applications to the list, see [“Adding New Applications to the Applications List” on page 120](#).

Configuring Basic Outbound QoS

The following example demonstrates how you can configure basic outbound QoS.

To configure QoS using the basic outbound QoS mode

1. Select Configure > Networking > Outbound QoS (Basic).
2. Enable QoS and set the link speed.
3. Verify the global application list.

Figure 4-6. The Global Application List

Configure > Networking > Outbound QoS (Basic) ?

QoS Settings

☒ Enable QoS Shaping
 WAN Bandwidth: kbps

☐ Enable QoS on **primary**

☒ Enable QoS on **wan0_0**

☐ Enable Local WAN Oversubscription

☐ Enable QoS Marking

Global DSCP:

Apply

Sites Applications Service Policies

+ Add Application - Remove Application ↑↓ Move Application

<input type="checkbox"/>	Order ↑↓	Name ↑↓	Service Class ↑↓	DSCP ↑↓
<input type="checkbox"/>	1	ICMP	Business-Critical	from Class
<input type="checkbox"/>	2	ICA	Business-Critical	from Class
<input type="checkbox"/>	3	CIFS	Normal	from Class
<input type="checkbox"/>	4	NFS	Normal	from Class
<input type="checkbox"/>	5	iTunes	Low-Priority	from Class
<input type="checkbox"/>	6	RDP	Business-Critical	from Class
<input type="checkbox"/>	7	DNS	Business-Critical	from Class
<input type="checkbox"/>	8	Facebook	Low-Priority	from Class

4. Adjust or create *service policies*.

A service policy allocates a percentage of the bandwidth to the different QoS classes. The DSCP or IP TOS value set (in this example set to Reflect) is shown with the allocated bandwidth.

Figure 4-7. Confirm That Service Policies Are Suitable for the Remote Offices

Sites Applications Service Policies							
+ Add Service Policy - Remove Service Policy							
<input type="checkbox"/>	Name ↑↓	Realtime	Interactive	Business-Critical	Normal	Low-Priority	Best-Effort
<input type="checkbox"/>	Large_Office	10-100% Reflect	10-100% Reflect	20-100% Reflect	50-100% Reflect	9-100% Reflect	
<input type="checkbox"/>	Larger_Office	20-100% Reflect	20-100% Reflect	20-100% Reflect	20-100% Reflect	19-100% Reflect	
<input type="checkbox"/>	Medium_Office	10-100% Reflect	20-100% Reflect	20-100% Reflect	40-100% Reflect	9-100% Reflect	
<input type="checkbox"/>	Small_Office	20-100% Reflect	20-100% Reflect	30-100% Reflect	20-100% Reflect	9-100% Reflect	
<input type="checkbox"/>	Smaller_Office	1-100% Reflect	1-100% Reflect	40-100% Reflect	40-100% Reflect	17-100% Reflect	

5. Associate remote sites with service policies.

Basic outbound QoS introduces the concept of a *site*. A site is a logical grouping of subnets that simplifies the configuration of the QoS rules. The advantage of using sites is more obvious when configuring QoS rules in the advanced outbound QoS rules. For details, see [“Advanced Outbound QoS” on page 88](#).

Figure 4-8. Associate Remote Offices With Different Service Policies

Sites Applications Service Policies					
+ Add Site - Remove Site ↑↓ Move Site					
<input type="checkbox"/>	Order ↑↓	Name ↑↓	Remote Subnets ↑↓	Service Policy ↑↓	Remote Bandwidth
<input type="checkbox"/>	1	London	10.22.1.0/24	Small_Office	2048.0
<input type="checkbox"/>	2	Paris	10.22.2.0/24	Large_Office	5120.0
<input type="checkbox"/>	3	Default-Site	0.0.0.0/0	Medium_Office	20.00

Figure 4-7 shows that by default the Medium Office profile is associated with the *Default-Site* covering all the remote locations. Default-Site is a catch-all site and it has a subnet of 0.0.0.0/0. Riverbed recommends that you create a new site.

For more details on basic outbound QoS configuration, see [“Configuring QoS on the Branch Office Steelhead Appliance” on page 119](#).

Basic Outbound QoS Mode Restrictions

Because basic outbound QoS is intended for simple configurations, you must ensure that yours meets certain basic conditions:

- You cannot add or delete classes. There are six predefined classes all using SFQ as the queue type.
- All interfaces have the same link rate (bandwidth).

Due to these basic outbound QoS restrictions, the following configurations require advanced outbound QoS:

- MX-TCP (requires mxtcp queue)

- Application priority for Citrix ICA traffic (requires packet-order queue)
- WAN links with different bandwidth: for example, there might be a 2 Mbps MPLS link with a 1 Mbps ADSL backup

WAN Oversubscription

Your network might include sites with pipes that collectively exceed the available bandwidth of the WAN uplink interface in the data center. For example, assume you have a data center uplink that is 45 Mbps and you have three remote offices with 20 Mbps circuit each. Without WAN oversubscription, the QoS configuration requires the sum of all uplink bandwidths to the remote sites to be less than or equal to the uplink bandwidth of the site where you perform QoS configuration. In other words, it is necessary to manually calculate the scaling ratio.

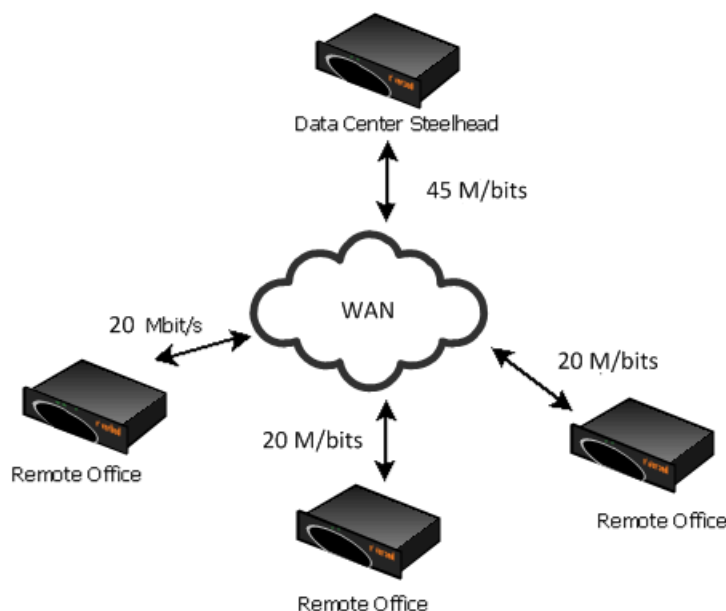
In basic outbound QoS mode, you can configure the remote sites so that the sum of their bandwidth exceeds the WAN uplink speed. This is known as *WAN oversubscription*, and it is available only in basic outbound QoS environments. You can specify the remote site bandwidth in bits per second instead of percentages relative to the WAN uplink bandwidth.

Use a *scale factor* to scale down bandwidth allocations (minimum bandwidth guarantee) accordingly.

Figure 4-9 shows the following example:

- WAN Uplink: 45 Mbps
- Three Remote Sites: 20 Mbps each
- Scale Factor: $45 / 60 = 0.75$
- Remote Site Bandwidth Guarantee: $20 \text{ Mbps} * 0.75 = 15 \text{ Mbps}$
- Remote Site Max Bandwidth: 20 Mbps

Figure 4-9. WAN Oversubscription



If all three offices are active at the same time, then each office is guaranteed 15 Mbps of bandwidth. If one of the offices is inactive, then the other two offices still receive the guaranteed minimum bandwidth of 15 Mbps. There is an excess bandwidth of 10 Mbps that is shared between the two active sites with equal weight. Therefore, both active sites receive full 20 Mbps allocation when the data center uplink is not fully used.

Note: *Scale factor* is calculated automatically based on the WAN link bandwidth as well as the remote link bandwidth. You can not manually adjust this setting.

Advanced Outbound QoS

Advanced outbound QoS mode offers a more granular approach to QoS configuration. Advanced outbound QoS mode contains all of the QoS features in releases earlier than RiOS v6.5. This section includes the following topics:

- [“QoS Classes” on page 88](#)
- [“Choosing a QoS Enforcement System” on page 91](#)
- [“QoS Class Parameters” on page 92](#)
- [“QoS Rules” on page 94](#)

QoS Classes

This section describes Riverbed QoS classes. This section includes the following topics:

- [“Hierarchical Mode” on page 88](#)
- [“Flat Mode” on page 91](#)

There following QoS classes are always present on the Steelhead appliance:

- **Root class** - The root class constrains the total outbound rate of traffic leaving the Steelhead appliance to the configured, per-link WAN bandwidth. This class is not configured directly, but is created by default when you enable QoS classification and enforcement on the Steelhead appliance.
- **Built-in default class** - The QoS scheduler applies the built-in default class constraints and parameters on traffic not otherwise placed in a class by the configured QoS rules. You must adjust the minimum bandwidth value for the default class to the appropriate value for your deployment. The default class cannot be deleted. It has a bandwidth of 10% that can be adjusted as low as 0%.

QoS classes are configured in one of two different modes: flat mode or hierarchical mode. The difference between the two modes primarily consists of how QoS classes are created.

Hierarchical Mode

In hierarchical mode, you can create QoS classes as children of QoS classes other than the root class. This allows you to create overall parameters for a certain traffic type, and specify parameters for subtypes of that traffic. You can create any number of QoS classes levels—there is no enforced limit.

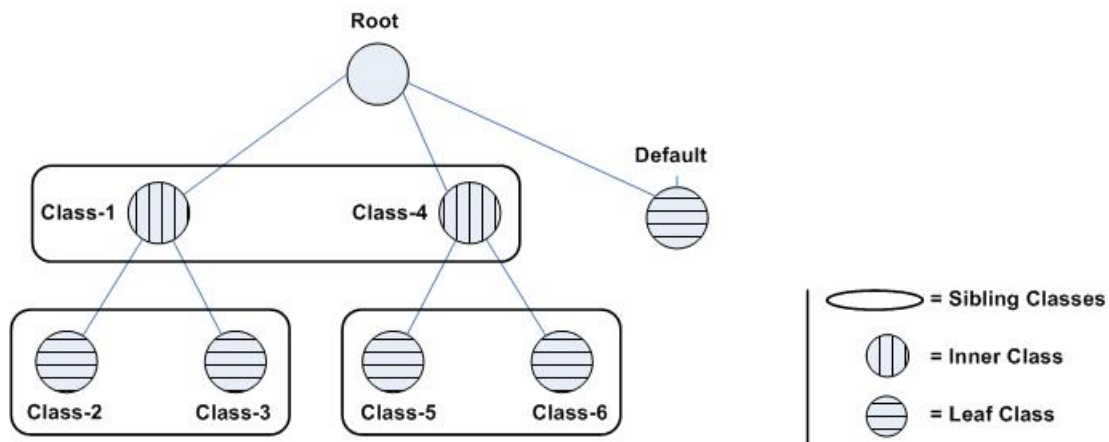
In hierarchical mode, the following relationships exist between QoS classes:

- **Sibling classes** - Classes that share the same parent class.

- **Leaf classes** - Classes at the bottom of the class hierarchy.
- **Inner classes** - Classes that are neither the root class nor leaf classes.

In hierarchical mode, QoS rules can only specify leaf classes as targets for traffic. [Figure 4-10](#) shows the hierarchical mode structure and the relationships between the QoS classes.

Figure 4-10. Hierarchical Mode Class Structure



QoS controls the traffic of hierarchical QoS classes in the following manner:

- QoS rules assign active traffic to leaf classes.
- The QoS scheduler:
 - applies active leaf class parameters to the traffic.
 - applies parameters to inner classes that have active leaf class children.
 - continues this process up the class hierarchy.
 - constrains the total output bandwidth to the WAN rate specified on the root class.

The following examples show how class hierarchy controls traffic.

[Figure 4-11](#) shows six QoS classes. The root and default QoS classes are built-in and are always present. This example shows the QoS class hierarchical mode structure.

Figure 4-11. Example of QoS Class Hierarchy

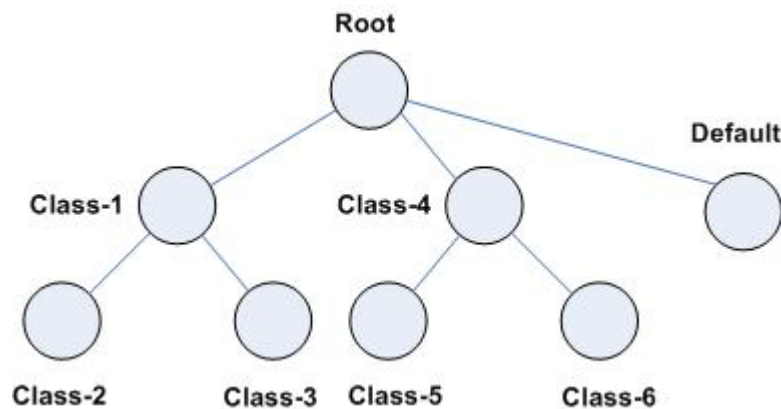
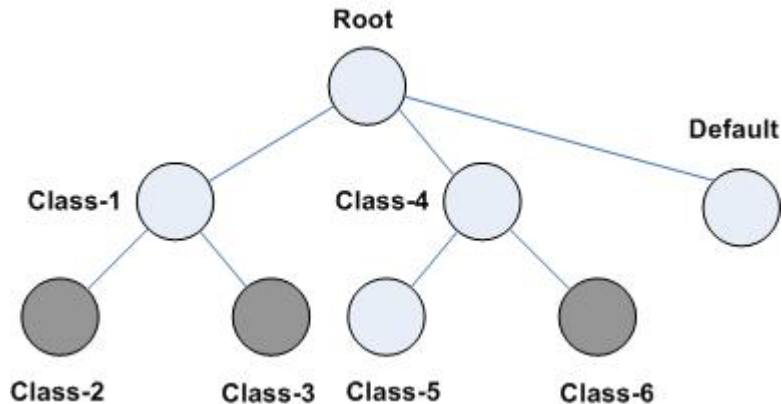


Figure 4-12 shows there is active traffic beyond the overall WAN bandwidth rate. This example shows a scenario in which the QoS rules place active traffic into three QoS classes: Classes 2, 3, and 6.

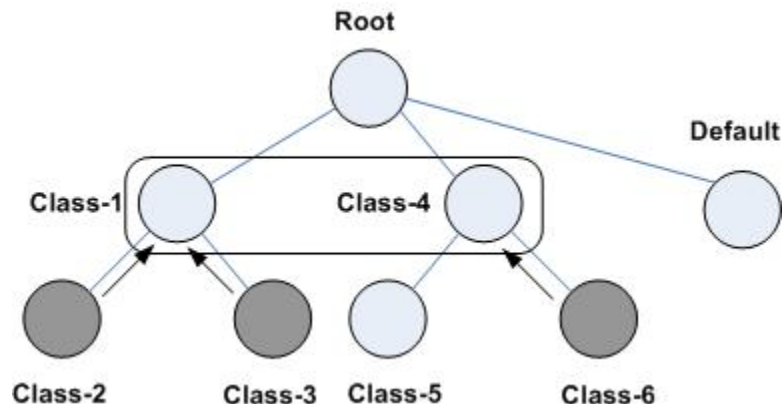
Figure 4-12. QoS Classes 2, 3, and 6 Have Active Traffic



Riverbed QoS rules place active traffic into QoS classes in the following manner. In the following order, the QoS scheduler:

1. applies the constraints for the lower leaf classes.
 2. applies bandwidth constraints to all leaf classes. The QoS scheduler awards minimum guarantee percentages among siblings, after which the QoS scheduler awards excess bandwidth, after which the QoS scheduler applies upper limits to the leaf class traffic.
 3. applies latency priority to the leaf classes. For example, if class 2 is configured with a higher latency priority than class 3, the QoS scheduler gives traffic in class 2 the chance to be transmitted before class 3. Bandwidth guarantees still apply for the classes.
 4. applies the constraints of the parent classes. The QoS scheduler treats the traffic of the children as one traffic class. The QoS scheduler uses class 1 and class 4 parameters to determine how to treat the traffic.
- Figure 4-13 shows the following points:
- Traffic from class 2 and class 3 is logically combined and treated as if it were class 1 traffic.
 - Because class 4 only has active traffic from class 6, the QoS scheduler treats the traffic as if it were class 4 traffic.

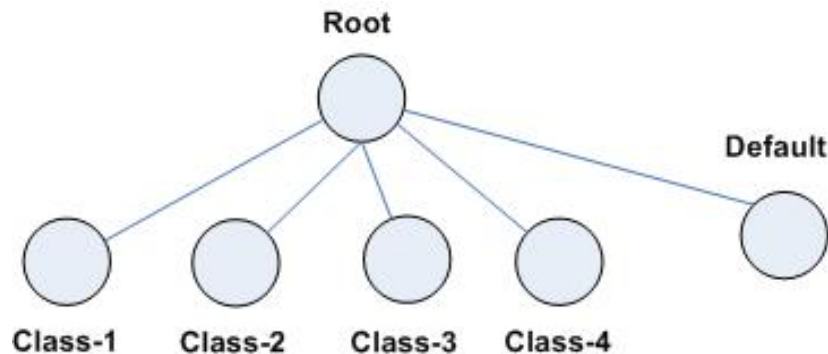
Figure 4-13. How the QoS Scheduler Applies Constraints of Parent Class to Child Classes



Flat Mode

In flat mode, you cannot define parent classes. All of the QoS classes you create have the same parent class, the root class. All of the QoS classes you create are siblings. [Figure 4-14](#) shows the flat mode structure.

Figure 4-14. Flat Mode Class Structure



The QoS scheduler treats QoS classes in flat mode the same way that it does in hierarchical mode. However, only a single class level is defined. QoS rules place active traffic into the leaf classes. Each active class has its own QoS rule parameters, which the QoS scheduler applies to traffic.

Choosing a QoS Enforcement System

The appropriate QoS enforcement system to use depends on the location of WAN bottlenecks for traffic leaving the site.

The following model is typically used for implementing QoS:

- A site that acts as a data server for other locations, such as a data center or regional hub, typically uses hierarchical mode. The first level of classes represents remote sites, and those remote site classes have child classes that either represent application types, or are indirectly connected remote sites.
- A site that typically receives data from other locations, such as a branch site, typically uses flat mode. The classes represent different application types.

For example, suppose you have a network with ten locations, and you want to choose the correct mode for Site 1. Traffic from site 1 normally goes to two other sites: sites 9 and 10. If the WAN links at sites 9 and 10 are at a higher bandwidth than the link at site 1, the WAN bottleneck rate for site 1 is always the link speed for site 1. In this case, you can use flat mode to enforce QoS at site 1, because the bottleneck that needs to be managed is the link at site 1. In flat mode, the parent class for all created classes is the root class that represents the WAN link at site 1.

In the same network, site 10 sends traffic to sites 1 through 8. Sites 1 through 8 have slower bandwidth links than site 10. Because the traffic from site 10 faces multiple WAN bottlenecks (one at each remote site), you configure hierarchical mode for site 10.

Note: Changing the QoS enforcement mode while QoS is enabled can cause disruption to traffic flowing through the Steelhead appliance. Riverbed recommends that you configure QoS while the QoS functionality is disabled and only enable it after you are ready for the changes to take effect.

QoS Class Parameters

This section describes QoS class parameters and includes the following topics:

- [“Per-Class Parameters” on page 92](#)
- [“QoS Class Latency Priorities” on page 92](#)
- [“QoS Queue Types” on page 93](#)
- [“MX-TCP” on page 93](#)

Per-Class Parameters

The QoS scheduler uses the per-class configured parameters to determine how to treat traffic belonging to the QoS class. The per-class parameters are:

- **Latency priority** - There are six QoS class latency priorities.
For details, see [“QoS Class Latency Priorities” on page 92](#).
- **Queue Types** - For details, see [“QoS Queue Types” on page 93](#).
- **Guaranteed minimum bandwidth** - When there is bandwidth contention, specifies the minimum amount of bandwidth as a percentage of the parent class bandwidth. The QoS class might receive more bandwidth if there is unused bandwidth remaining. In hierarchical mode, excess bandwidth is allocated based on the relative ratios of guaranteed bandwidth. The total minimum guaranteed bandwidth of all QoS classes must be less than or equal to 100% of the parent class. You can adjust the value as low as 0%.
- **Link share weight** - This applies to flat mode only. Specifies how excess bandwidth is allocated among sibling classes. In flat QoS, link share does not depend on the guaranteed minimum bandwidth. By default, all link shares are equal. QoS classes with a larger link-share weight are allocated more of the excess bandwidth than QoS classes with a lower link share weight. The link share weight does not apply to hierarchical QoS because hierarchical QoS allocates excess bandwidth based on the minimum guarantee for each class.
- **Maximum bandwidth** - Specifies the maximum allowed bandwidth a QoS class receives as a percentage of the parent class guaranteed bandwidth. The upper bandwidth limit is applied even if there is excess bandwidth available. The upper bandwidth limit must be greater than or equal to the minimum bandwidth guarantee for the class. The smallest value you can assign is 0.01%.
- **Connection limit** - Specifies the maximum number of optimized connections for the QoS class. When the limit is reached, all new connections are passed through unoptimized. In hierarchical mode, a parent class connection limit does not affect its child. Each child-class optimized connection is limited by the connection limit specified for its class. For example, if B is a child of A, and the connection limit for A is set to 5, although the connection limit for A is set to 10, the connection limit for B is 10. Connection limit is supported only in in-path configurations. Connection limit is not supported in out-of-path or virtual-in-path configurations.

RiOS does not support a connection limit assigned to any QoS class that is associated with a QoS rule with a AFE component. An AFE component consists of a Layer-7 protocol specification. RiOS cannot honor the class connection limit because the QoS scheduler might subsequently reclassify the traffic flow after applying a more precise match using AFE identification.

QoS Class Latency Priorities

Latency priorities indicate how delay-sensitive a traffic class is. A latency priority does not control how bandwidth is used or shared among different QoS classes. You can assign a QoS class latency priority when you create a QoS class or modify it later.

Riverbed QoS has six QoS class latency priorities. The following table summarizes the QoS class latency priorities in descending order.

Latency Priority	Example
Real time	VoIP, video conferencing
Interactive	Citrix, RDP, telnet, and ssh
Business critical	Thick client applications, ERPs, CRMs
Normal priority	Internet browsing, file sharing, email
Low priority	FTP, backup, replication, and other high-throughput data transfers; recreational applications such as audio file sharing
Best effort	lowest priority

Typically, applications such as VoIP and video conferencing are given real-time latency priority, although applications that are especially delay-insensitive, such as backup and replication, are given low latency priority.

Important: The latency priority describes only the delay sensitivity of a class, not how much bandwidth it is allocated, nor how important the traffic is compared to other classes. Therefore, it is common to configure low latency priority for high-throughput, delay-insensitive applications such as ftp, backup, and replication.

QoS Queue Types

Each QoS class has a configured queue type parameter. There following types of parameters available:

- **Stochastic Fairness Queuing (SFQ)** - Determines Steelhead appliance behavior when the number of packets in a QoS class outbound queue exceeds the configured queue length. When SFQ is used, packets are dropped from within the queue, among the present traffic flows. SFQ ensures that each flow within the QoS class receives a fair share of output bandwidth relative to each other, preventing bursty flows from starving other flows within the QoS class. SFQ is the default queue parameter.
- **First-in, First-Out (FIFO)** - Determines Steelhead appliance behavior when the number of packets in a QoS class outbound queue exceeds the configured queue length. When FIFO is used, packets received after this limit is reached are dropped, hence the first packets received are the first packets transmitted.
- **Packet-order** - Protects the TCP stream order by keeping track of flows that are currently inside the packet-shaping infrastructure. Packet-order protection allows only one packet from each flow into the HFSC traffic shaper at a time. The backlog for each flow stores the packets from the flow in order until the packet inside the HFSC infrastructure is dequeued for delivery to the network interface. This packet-order protection works for both TCP and UDP streams. Select this queue with the Citrix QoS classes for best performance.

MX-TCP

MX-TCP is a QoS class queue parameter as well, but with very different use cases than the other queue parameters. MX-TCP also has secondary effects that you must understand before you configure it.

When optimized traffic is mapped into a QoS class with the MX-TCP queuing parameter, the TCP congestion control mechanism for that traffic is altered on the Steelhead appliance. The normal TCP behavior of reducing the outbound sending rate when detecting congestion or packet loss is disabled, and the outbound rate is made to match the minimum guaranteed bandwidth configured on the QoS class.

You can use MX-TCP to achieve high throughput rates even when the physical medium carrying the traffic has high loss rates. For example, a common usage of MX-TCP is for ensuring high throughput on satellite connections where no lower-layer loss recovery technique is in use.

Another usage of MX-TCP is to achieve high throughput over high-bandwidth, high-latency links, especially when intermediate routers do not have properly tuned interface buffers. Improperly tuned router buffers cause TCP to perceive congestion in the network, resulting in unnecessarily dropped packets, even when the network can support high throughput rates.

MX-TCP is incompatible with the AFE. A traffic flow cannot be classified as MX-TCP and then subsequently classified in a different queue. This reclassification can happen if there is a more exact match of the traffic.

You must ensure the following when you enable MX-TCP:

- The QoS rule for MX-TCP is at the top of QoS rules list.
- The rule does not use AFE identification.
- Use MX-TCP for optimized traffic only.

Important: Use caution when you specify MX-TCP. The outbound rate for the optimized traffic in the configured QoS class immediately increases to the specified bandwidth, and does not decrease in the presence of network congestion. The Steelhead appliance always tries to transmit traffic at the specified rate. If no QoS mechanism (either parent classes on the Steelhead appliance or another QoS mechanism in the WAN or WAN infrastructure) is in use to protect other traffic, that other traffic can be impacted by MX-TCP not adjusting to share bandwidth.

In RiOS v8.0 or later, you can configure MX-TCP to scale beyond the specified minimum bandwidth up to a specific maximum. This allows MX-TCP to use available bandwidth during non-peak hours. During traffic congestion, MX-TCP scales back to the configured minimum bandwidth. For details on configuring MX-TCP before RiOS v8.5, see earlier versions of the *Steelhead Appliance Deployment Guide* and *Riverbed Deployment Guide* on the Riverbed Support site at <https://support.riverbed.com>.

You can configure a specific rule for an MX-TCP class for packet-mode UDP traffic. An MX-TCP rule for packet-mode UDP traffic is useful for UDP bulk transfer and data-replication applications (for example, Aspera and Veritas Volume Replicator).

Packet-mode traffic matching any non-MX-TCP class is classified into the default class because QoS does not support packet-mode optimization.

For more information about MX-TCP as a transport streaming lining mode, see [“Overview of Transport Streamlining” on page 9](#). For an example of how to configure QoS and MX-TCP, see [“Configuring QoS and MX-TCP” on page 135](#).

QoS Rules

QoS rules map different types of network traffic to QoS classes. After you define a QoS class, you can create one or more QoS rules to assign traffic to it. QoS rules can match traffic based on:

- a source address, port, or subnet.
- a destination address, port, or subnet.
- the IP protocol in use: TCP, UDP, GRE, or all.
- whether the traffic is optimized.
- a VLAN tag.
- a DSCP/IP TOS value.

- payload.
- Citrix application-specific bits in the payload. You can examine Citrix payloads for the application priority.
- PC-over-IP (PCoIP) application-specific bits in the payload. You can examine PCoIP payloads for the application priority.
- SnapMirror application-specific bits in the payload, allowing you to classify and DSCP-mark traffic coming from particular volumes and qtrees.

For QoS rules that do not use AFE identification, the rules are processed in the order they are shown on the QoS Classification page of the Management Console. The first matching rule determines which QoS class the traffic is assigned to. A QoS class can have many rules assigning traffic to it.

For details on QoS rules using the AFE, see the configuration example in [Figure 4-15](#).

In hierarchical mode, QoS rules can only be defined for, and map traffic to, the leaf classes. You cannot associate QoS rules to inner classes.

In releases earlier than RiOS v6.5, the QoS rules were a flat structure. For rules belonging to the same location, you must configure the destination subnet in each rule. RiOS v6.5 or later introduces the concept of sites. Sites describe the logical subnets of a remote location that are only defined once. A site streamlines the layout of the QoS rules, making it easier to understand.

A default QoS rule always exists at the end of the QoS rule list and cannot be deleted. The default rule is used for traffic that does not match any rules in the QoS rule list. The default rule assigns this traffic to the built-in default QoS class.

Consider Layer-7 QoS rules in the QoS rule configuration shown in [Figure 4-15](#).

Figure 4-15. QoS Rule Configuration

QoS Sites and Rules:					
+ Add a Site or Rule		- Remove Sites or Rules		⇅ Move...	
<input type="checkbox"/>	Order	Name	Service Class	DSCP	Application
<input type="checkbox"/>	Site 1	▼ Paris			
	default	🔍 default	Paris\$\$Low-Priority	Reflect	
<input type="checkbox"/>	Site 2	▼ London			
<input type="checkbox"/>	1	🔍 QoS Rule 1	London\$\$Normal	from Class	HTTP Domain: www.acme.com
<input type="checkbox"/>	2	🔍 QoS Rule 2	London\$\$Low-Priority	from Class	HTTP
	default	🔍 default	London\$\$Low-Priority	Reflect	
	Site 3	▶ Default-Site			

Paris has the subnet 10.1.0.0/16 and London has the subnet 10.0.0.0/15.

Figure 4-16. Paris Office

QoS Sites and Rules:

+ Add a Site or Rule - Remove Sites or Rules ↑↓ Move...

Order	Name	Service Class
	▼ Paris	

Subnets:

10.1.0.0/16

Apply

Figure 4-17. London Office

QoS Sites and Rules:

+ Add a Site or Rule - Remove Sites or Rules ↑↓ Move...

Order	Name	Service Class
	▼ Paris	
default	default	Default-Site\$\$Low-Priority
	▼ London	

Subnets:

10.0.0.0/15

Apply

The London site has two HTTP rules. The first rule matches on HTTP and a specific domain name. Traffic matching this rule is placed in the normal latency priority class. The second rule is a generic rule matching all HTTP traffic. Traffic matching this rule is placed in the low latency priority class.

In the following scenario, a client opens a connection to a Web server with IP address 10.0.1.1:

- The first packet from the client to server is a HTTP "GET / HTTP1.1". It does not contain the domain name of the host. This packet matches the header (the IP addresses and port number) rule of rule number 1 (Figure 4-18). When the QoS engine tries to perform a match on the header, the AFE informs the QoS engine that it requires additional data. The QoS engine considers the packet as not matching rule number 1 and moves down the list. As the QoS engine moves down the list, the packet matches the header rule and the AFE rule for rule number 2 (Figure 4-19). In this case, the first packet is placed in the Low latency priority class. Normally, the result is saved in a table so that further packets from the same flow are not checked against the table. Because there is a rule before rule number 2 asking for more data, the second packet runs through the rules table also.

- The second packet from the client contains the domain name of the host and it matches rule number 1. The result of this match is saved in a table, and subsequent packets from the same flow do not need to consult the QoS rules again.

Figure 4-18. QoS Rule 1

QoS Sites and Rules:

[+ Add a Site or Rule](#) [- Remove Sites or Rules](#) [⇅ Move...](#)

	Order	Name	Service Class
<input type="checkbox"/>	Site 1	► Q Paris	
<input type="checkbox"/>	Site 2	▼ Q London	
<input type="checkbox"/>	1	QoS Rule 1	London\$\$Normal

Name:

Description:

Parent Site:

Insert Rule At:

For Traffic with the Following Characteristics:

Local Subnet or [Host Label](#): Port or [Port Label](#):

Remote Subnet or [Host Label](#): Port or [Port Label](#):

Protocol:

VLAN Tag ID:

DSCP:

Traffic Type:

Application:

Domain Name: Use * to indicate wildcards.

Relative Path: Use * to indicate wildcards.

Apply these QoS Settings:

Service Class:

DSCP:

Apply these Path Selections:

[Path](#) preference order (only one path will be used):

Path 1: DSCP:

Path 2: DSCP:

Path 3: DSCP:

If no configured path is up:

☒ Relay traffic from the interface normally.

☐ Drop traffic.

Figure 4-19. QoS Rule 2

QoS Sites and Rules:

+ Add a Site or Rule - Remove Sites or Rules ⇅ Move...

Order	Name	Service Class
	Site 1	Q Paris
	Site 2	Q London
1	QoS Rule 1	London\$\$Normal
2	QoS Rule 2	London\$\$Low-Priority

Name:

Description:

Parent Site:

Insert Rule At:

For Traffic with the Following Characteristics:

Local Subnet or Host Label: Port or Port Label:

Remote Subnet or Host Label: Port or Port Label:

Protocol:

VLAN Tag ID:

DSCP:

Traffic Type:

Application:

Domain Name: Use * to indicate wildcards.

Relative Path: Use * to indicate wildcards.

Apply these QoS Settings:

Service Class:

DSCP:

Apply these Path Selections:

Path preference order (only one path will be used):

Path 1: DSCP:

Path 2: DSCP:

Path 3: DSCP:

If no configured path is up:

☒ Relay traffic from the interface normally.

☐ Drop traffic.

Inbound QoS

RiOS v7.0.1 or later includes inbound QoS. Inbound QoS enables you to allocate bandwidth and prioritize traffic flowing from the WAN into the LAN network behind the Steelhead appliance. This provides the benefits of QoS for environments that cannot meet their QoS requirements with outbound QoS. This section includes the following topics:

- [“Inbound QoS Limitations” on page 99](#)
- [“Inbound QoS Limits” on page 100](#)

Reasons to configure inbound QoS include the following:

- Many business applications, such as VoIP and desktop video conferencing, now run over any-to-any mesh topologies.

The traffic generated by these applications typically travels directly from one branch office to another; for example, if a user in Branch Office A calls a user in Branch Office B, the VoIP call is routed directly, without having to traverse the data center. This traffic might compete with other traffic that is coming from the data center, or from other sites, but it bypasses any QoS that is deployed at those sites. As a result, there is no network location from which you can use outbound QoS to control all incoming traffic going to Branch Office B. The only place where you can control all incoming traffic is at the branch itself. You can use inbound QoS at Branch Office B to guarantee bandwidth for critical applications and slow down traffic from other, less critical applications.

- Software as a Service (SaaS) applications and public cloud services that are accessed over the Internet.

These applications compete with recreational Internet traffic for bandwidth at the branch office. When users watch online videos or browse social networking sites, business applications can struggle to get the resources they need. With inbound QoS, you can ensure that business applications have enough room to get through.

Although inbound configuration is separate from the outbound configuration, you use inbound QoS the same way you use outbound QoS—to prioritize types of traffic using rules and classes. You define the applications on the local network and then create their corresponding shaping policies.

For details on how to configure inbound QoS, see *Steelhead Appliance Management Console User's Guide*.

Inbound QoS applies the HFSC shaping policies to the ingress traffic. This addresses environments in which bandwidth constraints exist at the downstream location. When this occurs, the downstream Steelhead appliance (where inbound QoS is enabled) dynamically communicates the bandwidth constraints to the client transmitting the traffic. The client slows down the throughput and the traffic adheres to the configured inbound QoS rule. Inbound QoS, just like outbound QoS, is not a dual ended Steelhead appliance solution. A single Steelhead appliance can control inbound WAN traffic on its own.

For details on the HFSC queuing technology, see [“Introduction to Riverbed QoS” on page 76](#) and the *Steelhead Appliance Management Console User's Guide*.

For more information about inbound QoS, see *Inbound QoS: A Riverbed Technical Whitepaper*.

Inbound QoS Limitations

The following limitations apply to inbound QoS traffic shaping:

- Unlike advanced outbound QoS, inbound QoS does not support hierarchical mode; it uses flat mode only. A deployment using inbound QoS typically does not require a hierarchical configuration to support multiple sites across the WAN. Deployment scenarios include configuring hierarchical mode using outbound QoS in the data center and configuring inbound QoS in the branch offices.
- If you use connection forwarding, inbound QoS does not classify and shape traffic it receives from a peer Steelhead appliance. In a connection forwarding environment, the traffic is classified and shaped by the first Steelhead appliance that detects the traffic.
- Inbound QoS is not fully compatible with RSP when one or more virtual machines are connected to the in-path dataflow. In such scenarios, all traffic that enters any virtual machine connected to the in-path dataflow falls into the default class.
- If you enable packet-mode optimization, the QoS scheduler places all of the accelerated traffic into the default class. All other traffic is placed in the proper class.
- You cannot configure inbound QoS in an out-of-path deployment on the auxiliary interface.
- Inbound QoS is not able to throttle flows such as MX-TCP and UDP bulk traffic flows; however, it does provide bandwidth and latency reservation for them.

Inbound QoS Limits

The following limits apply to inbound QoS traffic shaping.

- The maximum number of inbound QoS rules is 500.
- The maximum number of inbound QoS classes is 200.

The inbound QoS connection classification limit depends on the Steelhead appliance model. When the number of optimized connections exceeds the connection classification limit for the appliance model, all new connections are passed through unoptimized.

For outbound QoS limit recommendations, see [“Guidelines for the Maximum Number of QoS Classes, Sites, and Rules” on page 100](#).

Guidelines for the Maximum Number of QoS Classes, Sites, and Rules

The number of QoS classes, sites, and rules you can create on a Steelhead appliance depends on the appliance model number, the traffic flow, and other RiOS features you enable.

Important: If you are using a release previous to RiOS v8.5.1, some of the features described in the chapter might not be applicable. For details on QoS before RiOS v8.5.1, see earlier versions of the *Steelhead Appliance Deployment Guide* and *Riverbed Deployment Guide* on the Riverbed Support site at <https://support.riverbed.com>.

Riverbed recommends that you do not exceed the guidelines in this section.

RiOS v7.0.3 and later enforces a maximum configurable root bandwidth per appliance model. Bandwidth enforcement issues a warning when you configure the sum of the bandwidth interfaces as a value greater than the model-specific QoS limits. The limits were introduced in RiOS v6.5.4 and v7.0.1. The warning appears when you save a configured bandwidth limit that exceeds the supported limit. If you receive the warning, adjust the sum of the configured QoS interface values to be lower or equal to the model-specific bandwidth limit and save the configuration.

Riverbed strongly recommends you configure the bandwidth at or below the appliance limit, as problems arise when you exceed it. Upgrading a Steelhead appliance to RiOS v7.0.3 does not automatically fail when the configuration has a greater QoS bandwidth limit than the appliance supports. However, after the upgrade, RiOS begins enforcing the bandwidth limits and disallows any QoS configuration changes. This could result in having to reconfigure all QoS policies to accommodate the bandwidth limit.

In RiOS v8.5.1 and later, QoS limits are no longer enforced, with the exception of the QoS bandwidth. To restrict the possibility of exhausting the system resources, use these safeguard limits:

- Maximum number of total rules: 2000
- Maximum number of sites in basic outbound QoS mode: 100
- Maximum number of sites in advanced outbound QoS mode: 200
- Maximum number of subnets per site: 50

In RiOS v6.5 and later, QoS performance is based on the number of connections per second. The guidelines are as follows:

- Desktop models: 200 new connections per second
- 1U models: 500 new connections per second
- 3U models: 1,000 new connections per second

The following tables show recommendations for the number of QoS classes, sites, and rules for the xx50 series, the xx55 series, the xx60 series, and the VSH series.

Important: Exceeding the recommended limits can lead to severe delays when you change your Riverbed QoS configuration or boot the Steelhead appliance.

This table shows the bandwidth limits and the Riverbed recommended optimal performance guidelines for the Steelhead appliance xx50 models.

Steelhead Appliance xx50 Model	Maximum Configurable Root Bandwidth (Mbps)	Maximum Classes	Maximum Rules	Maximum Sites
150	4	150	150	15
250	4	150	150	15
550	10	250	250	25
1050	45	2,000	2,000	200
2050	100	2,000	2,000	200
5050	200	2,000	2,000	200
6050	310	2,000	2,000	200
7050	1,000	2,000	2,000	200

This table shows the bandwidth limits and the Riverbed recommended optimal performance guidelines for the Steelhead appliance xx55 (CX) models.

Steelhead Appliance CX Model	Maximum Configurable Root Bandwidth (Mbps)	Recommended Maximum Classes	Recommended Maximum Rules	Recommended Maximum Sites
CX255U	4	300	300	50
CX255L	12	300	300	50
CX255M	12	300	300	50
CX255H	12	300	300	50
CX555L	12	500	500	50
CX555M	20	500	500	50
CX555H	20	500	500	50
CX755L	45	1,000	1,000	100
CX755M	45	1,000	1,000	100
CX755H	45	1,000	1,000	100
CX1555L	100	2,000	2,000	200
CX1555M	100	2,000	2,000	200
CX1555H	100	2,000	2,000	200
CX5055L	No limit	2,000	2,000	200
CX5055M	No limit	2,000	2,000	200
CX5055H	No limit	2,000	2,000	200
CX7055L	No limit	2,000	2,000	200
CX7055M	No limit	2,000	2,000	200
CX7055H	No limit	2,000	2,000	200

This table shows the bandwidth limits and the Riverbed recommended optimal performance guidelines for the Steelhead appliance xx60 (EX) models.

Steelhead Appliance EX Model	Maximum Configurable Root Bandwidth (Mbps)	Recommended Maximum Classes	Recommended Maximum Rules	Recommended Maximum Sites
EX560	12 for G, L, M configurations 20 for H configuration	250	250	25
EX760	45	500	500	50
EX1160	100	500	500	50
EX1260	100	500	500	50
EX1360	100	500	500	100

This table shows the bandwidth limits and the Riverbed recommended optimal performance guidelines for the VSH (VCX) models.

Steelhead Appliance VCX Model	Maximum Configurable Root Bandwidth (Mbps)	Recommended Maximum Classes	Recommended Maximum Rules	Recommended Maximum Sites
VCX255U	4	150	150	25
VCX255L	12	150	150	25
VCX255M	12	150	150	25
VCX255H	12	150	150	25
VCX555L	12	250	250	25
VCX555M	20	250	250	25
VCX555H	20	250	250	25
VCX755L	45	500	500	50
VCX755M	45	500	500	50
VCX755H	45	500	500	50
VCX1555L	100	1000	1000	100
VCX1555M	100	1000	1000	100
VCX1555H	100	1000	1000	100

LAN Bypass

Virtual in-path network topologies in which the LAN-bound traffic traverses the WAN interface might require that you configure the Steelhead appliance to bypass LAN-bound traffic so that it is not subject to the maximum root bandwidth limit. Some deployment examples are WCCP or a WAN-side default gateway. The LAN bypass feature enables you to exempt certain subnets from QoS enforcement. You can configure LAN bypass on both inbound and outbound QoS.

For details on LAN bypass, see the *Steelhead Appliance Management Console User's Guide*.

Because xx50 Steelhead appliances follow performance guidelines but do not have enforced limits, you do not need to configure LAN bypass on these models.

QoS for IPv6

IPv6 traffic is not currently supported for QoS shaping or AFE-based classification. If you enable QoS shaping for a specific interface, all IPv6 packets for that interface are classified to the default class.

You can mark IPv6 traffic with an IP TOS value. You can also configure the Steelhead appliance to reflect an existing traffic class from the LAN side to the WAN side of the Steelhead appliance.

For more information about IPv6, see [“IPv6” on page 275](#).

QoS in Virtual In-Path and Out-of-Path Deployments

You can use QoS enforcement, on both inbound and outbound traffic, in virtual in-path deployments (for example, WCCP and PBR) and out-of-path deployments. In both of these types of deployments, you connect the Steelhead appliance to the network through a single interface: the WAN interface for WCCP deployments, and the primary interface for out-of-path deployments. You enable QoS for these types of deployments:

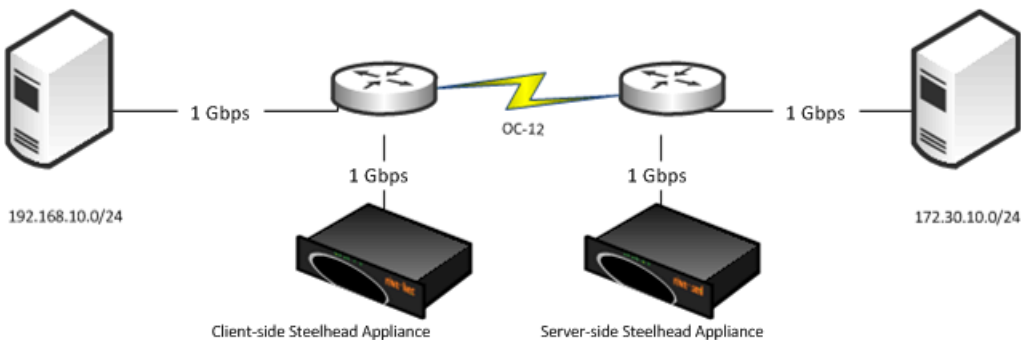
- Configure subnet side rules appropriately with LAN IP subnets on the LAN side. The subnet side rules define which part of the interface traffic belongs to the LAN side.

For details on subnet side rules, see the *Steelhead Appliance Management Console User's Guide*.

- Set the WAN throughput for the network interfaces to the total speed of the LAN and WAN interfaces, or to the speed of the local link, whichever number is lower.
- Configure QoS as if you have deployed the Steelhead appliance in a physical in-path mode.

Figure 4-20 shows bulky traffic that is unidirectional. The traffic is replicated and sent from the server in the data center to the disaster recovery site using the dedicated OC-12 link. Both Steelhead appliances are deployed in a virtual in-path deployment and the optimized traffic is placed in the MX-TCP class.

Figure 4-20. MX-TCP Example



As the optimized traffic is placed in the MX-TCP class, you must configure the advanced outbound QoS class. If necessary, perform the migration from basic outbound QoS to advanced outbound QoS before proceeding.

For an example configuration of QoS and MX-TCP on the client-side and server-side Steelhead appliance, see [“Configuring QoS and MX-TCP” on page 135](#).

For details on migration, see [“Migrating Between Basic and Advanced Outbound QoS Modes” on page 107](#).

QoS in Multiple Steelhead Appliance Deployments

You can use QoS when multiple Steelhead appliances are optimizing traffic for the same WAN link. In these cases, you must configure the QoS settings so that the Steelhead appliances share the available WAN bandwidth. For example, if traffic is to be load-balanced evenly across two Steelhead appliances, then the maximum WAN bandwidth configured for each Steelhead appliance is one-half of the total available WAN bandwidth. This scenario is often found in multiple Steelhead appliance data protection deployments.

For details on data protection deployments, see [“Designing for Scalability and High Availability” on page 351](#).

QoS and Multiple WAN Interfaces

You can enable QoS on WAN interfaces using different bandwidths. For example, you can connect interface wan0_0 to a 10-Mbps terrestrial link and connect interface wan0_1 to a 1-Mbps satellite backup link. You must use advanced outbound QoS because the interface speeds are different. Advanced outbound QoS supports different bandwidths for different interfaces; basic QoS does not. Basic QoS supports multiple interfaces, but they must all have the same bandwidth.

Keep in mind that the outbound WAN capacity limit is based per Steelhead appliance, regardless of the number of interfaces. For example, with a Steelhead appliance 1050-M, the outbound WAN capacity is limited to 10 Mbps of optimized traffic regardless of the number of interfaces installed or enabled for QoS.

The bandwidth and latency allocation are applied across all the interfaces. Continuing with the previous example, if you allocate 10% of the bandwidth for CIFS traffic, then that would be 10% of the 10-Mbps link and 10% of the 1-Mbps link.

QoS Enforcement Best Practices

Riverbed recommends the following actions to ensure optimal performance with the least amount of initial and ongoing configuration:

- Configure QoS while the QoS functionality is disabled and only enable it after you are ready for the changes to take effect.
- Ensure that Steelhead appliances at larger sites, such as data centers and regional hubs, use hierarchical mode.
- Ensure that Steelhead appliances at branch locations use flat mode.
- Increase the minimum guaranteed bandwidth and define the link share for the built-in default class.

The built-in default class is configured with a minimum guaranteed bandwidth of 0.01% and has no defined link share. These default values typically need to be altered. For example, in hierarchical mode, another QoS class allocated at the top-level with a minimum guaranteed bandwidth of 5% receives 500 times more of the link share than any QoS class found in the default class. A typical indication that the default class must be adjusted is when traffic that is not specified in the QoS classes (typical examples include Web browsing and routing updates) receives very little bandwidth during times of congestion.

- In hierarchical mode, if you are using a model in which the top-level QoS classes represent sites:
 - for each site, create a site-specific default class.

Create a QoS rule that comes after any other QoS rules that are specific to that site and that captures traffic to that site. Specify the per-site default class as the target so that no traffic is assigned to the built-in default class. Use the default class to dequeue important packets such as ARPs. You must use the default class to dequeue traffic.
 - configure the first-level classes to represent remote sites and the second-level classes to represent applications. For example, at data centers the first-level class represents regional hubs, and the second-level class represents indirectly connected sites.
- Be aware that flows can be incorrectly classified if there are asymmetric routes in the network in which you have enabled the QoS features.

Migrating Between Basic and Advanced Outbound QoS Modes

The basic outbound QoS mode and the advanced outbound QoS mode are mutually exclusive. Basic outbound QoS is the default on a Steelhead appliance without any previous QoS configurations. You can convert from basic outbound QoS to advanced outbound QoS, but you cannot do the reverse without losing the entire QoS configuration. After the basic outbound QoS configuration has been migrated to advanced outbound QoS, you must clear the entire QoS configuration to reverse back to basic outbound QoS mode.

Figure 4-21. Warning Message when Migrating from Basic Outbound QoS to Advanced Outbound QoS

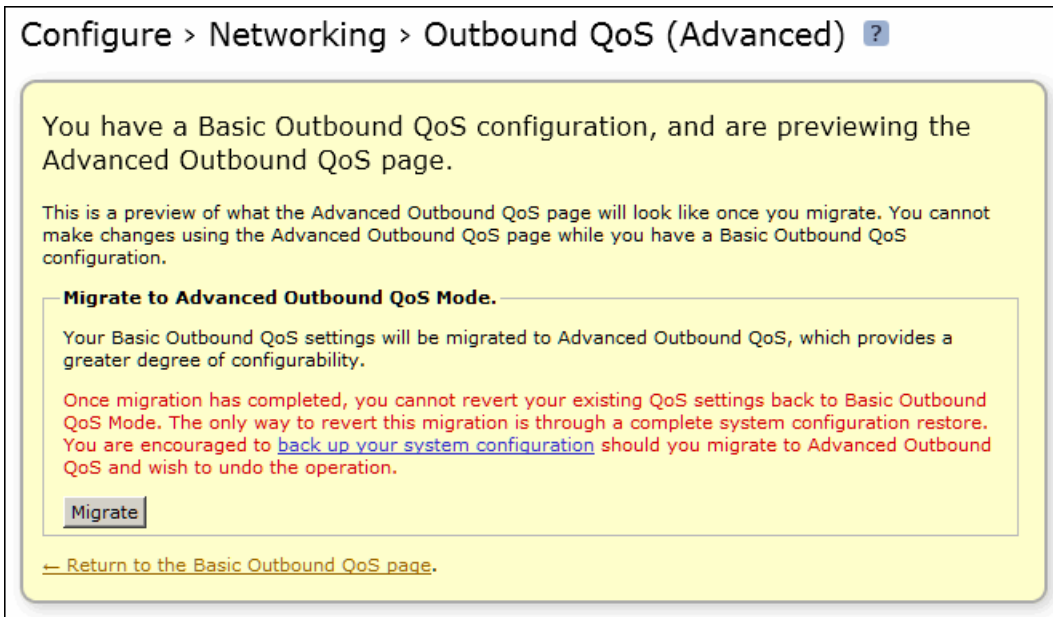
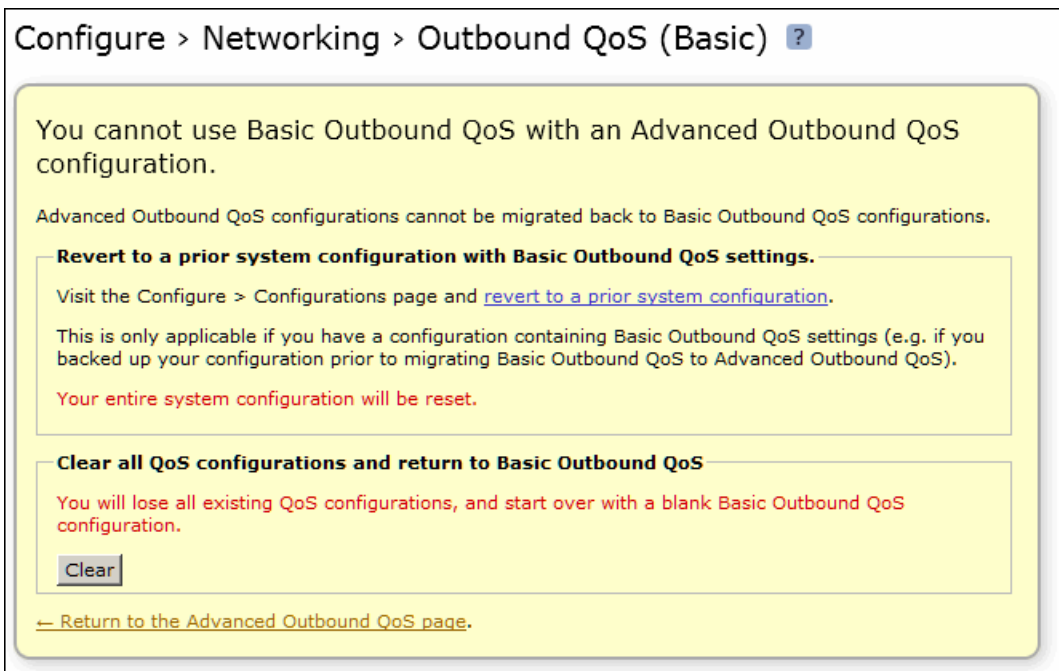


Figure 4-22. Warning Message when Migrating from Advanced Outbound QoS to Basic Outbound QoS



Upgrading to RiOS v6.5 or Later

When you upgrade a Steelhead appliance running releases earlier than RiOS v6.5 with QoS already configured, the upgrade process automatically places the QoS configuration in advanced outbound QoS mode to preserve all the QoS information. Switching to basic outbound QoS is only possible by clearing all the QoS configuration and starting from a clean QoS configuration.

For more details, see [“Migrating Between Basic and Advanced Outbound QoS Modes”](#) on page 107.

CHAPTER 5 QoS Configuration Examples

This chapter provides examples of QoS configurations. This chapter includes the following sections:

- [“Visualizing and Drawing Your QoS Configuration” on page 109](#)
- [“Configuring QoS Using Best Practices” on page 113](#)
- [“Configuring QoS Marking on Steelhead Appliances” on page 121](#)
- [“Configuring QoS for Citrix Traffic” on page 125](#)
- [“Configuring QoS and MX-TCP” on page 135](#)
- [“Creating Host Labels” on page 141](#)
- [“Configuring QoS for SSL Common Name Matching” on page 142](#)
- [“Configuring QoS for PCoIP” on page 143](#)
- [“Configuring QoS for SnapMirror” on page 145](#)

You can also use the Riverbed CLI to configure QoS. For detailed information about QoS commands, see the *Riverbed Command-Line Interface Reference Manual*.

You can use the CMC to enable QoS and to configure and apply QoS rules to multiple Steelhead appliances. For details, see the *Riverbed Central Management Console User’s Guide*.

Visualizing and Drawing Your QoS Configuration

This section shows how you can plan an advanced QoS, server-side Steelhead appliance configuration with simple drawings.

Visualizing and drawing your QoS configuration prior to implementation greatly enhances your understanding of your deployment strategy, because you are drawing a portrait of the traffic classification. Once you have completed this drawing, your ability to configure QoS is vastly improved. Using the QoS configuration drawing, you can easily see the relationships between the child and parent classes. Visualizing these relationships helps you to avoid missing important traffic on which you want to apply QoS.

This example has the following attributes:

- The number of remote sites, with a certain bandwidth to each
- The number of generic applications to classify

- A fixed amount of throughput for specific business applications
- Internet-bound traffic egress from the data center, which needs to be classified and the bandwidth controlled

To visualize and draw your QoS configuration

1. Draw a container, which represents the root class.

This container equals the total amount of allotted bandwidth. The root class represents the entire bandwidth you can divide among child classes. The root class bandwidth is directly equal to the configured-bandwidth value.

2. Draw another of container within the root class, that represents the total bandwidth that you can use at remote sites.

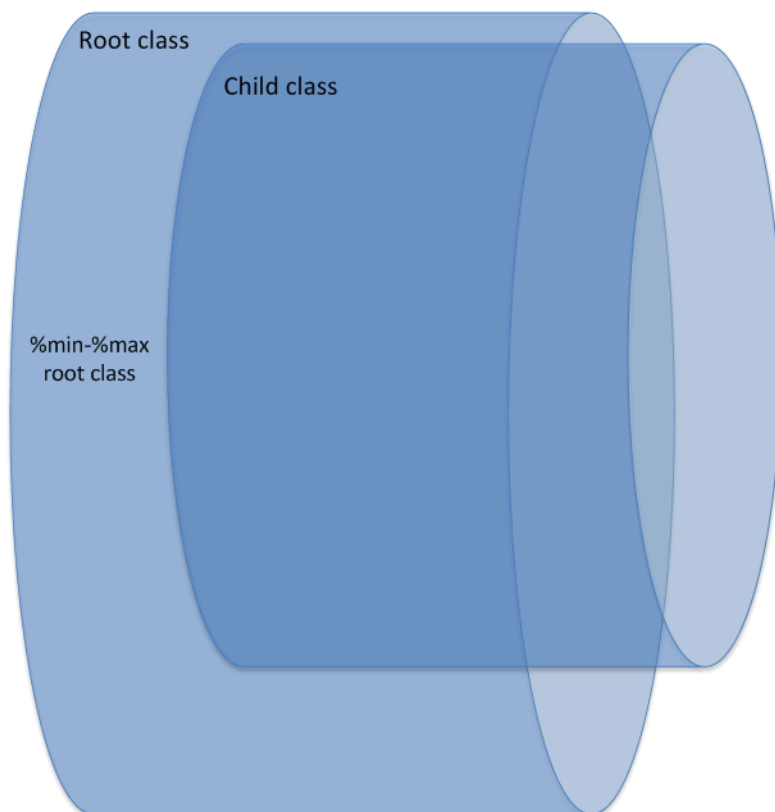
Each remote site container is representative of a child class on the Steelhead appliance. The class bandwidth value is represented by a percentage value based on the parent value. In this example, the parent is the root class.

The remote site container is the cumulative bandwidth dedicated for all of the remote sites. For example, if you have a DS-3 (45 Mbit) dedicated for all remote sites, then the percentage number should equal the calculation result from the main configured bandwidth. This configured percentage might not equal the combined bandwidth of all the remote sites—it is common to oversubscribe on WAN links.

For information about WAN oversubscription, see [“WAN Oversubscription” on page 87](#).

Your drawing should look something like [Figure 5-1](#).

Figure 5-1. Draw Containers for the Root and Child Class



3. Draw another container to represent a single remote site, including bandwidth percentage (Site A).

The percentage value represents the bandwidth allotment for this specific site. Remember that the maximum value should not exceed the remote site throughput capacity. The maximum defined value is directly related to the parent class bandwidth reservation. For example, if the parent can burst to 100%, and assuming that no other traffic is using the bandwidth, then a 100% child configuration enables the child class to take advantage of the maximum value (which might not be a desirable output).

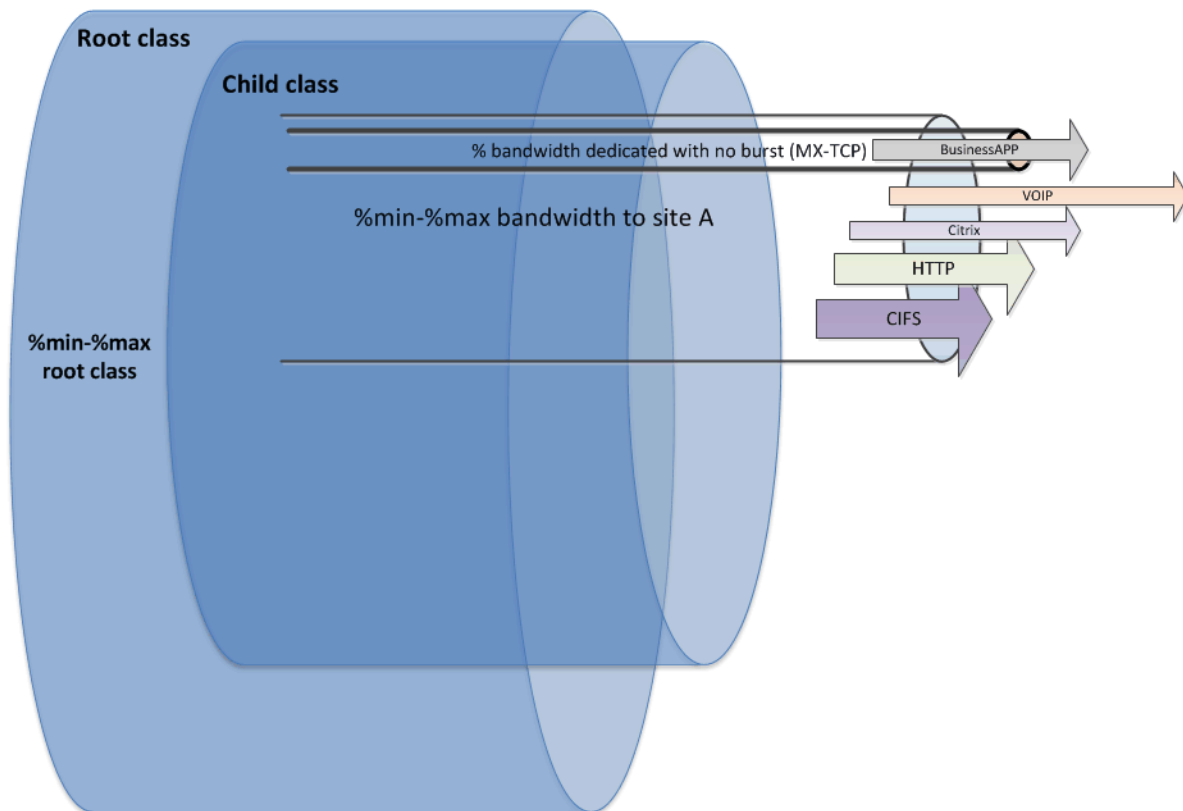
4. Inside of this remote site container, add your applications.

5. Assign a bandwidth percentage to each application.

Use arrow size to represent the throughput. The arrow can grow or shrink according to the minimum or maximum value. The size of the arrows in the drawing determines application priority in the physical configuration.

Your drawing should look something like [Figure 5-2](#).

Figure 5-2. Draw Total Bandwidth for Remote Sites

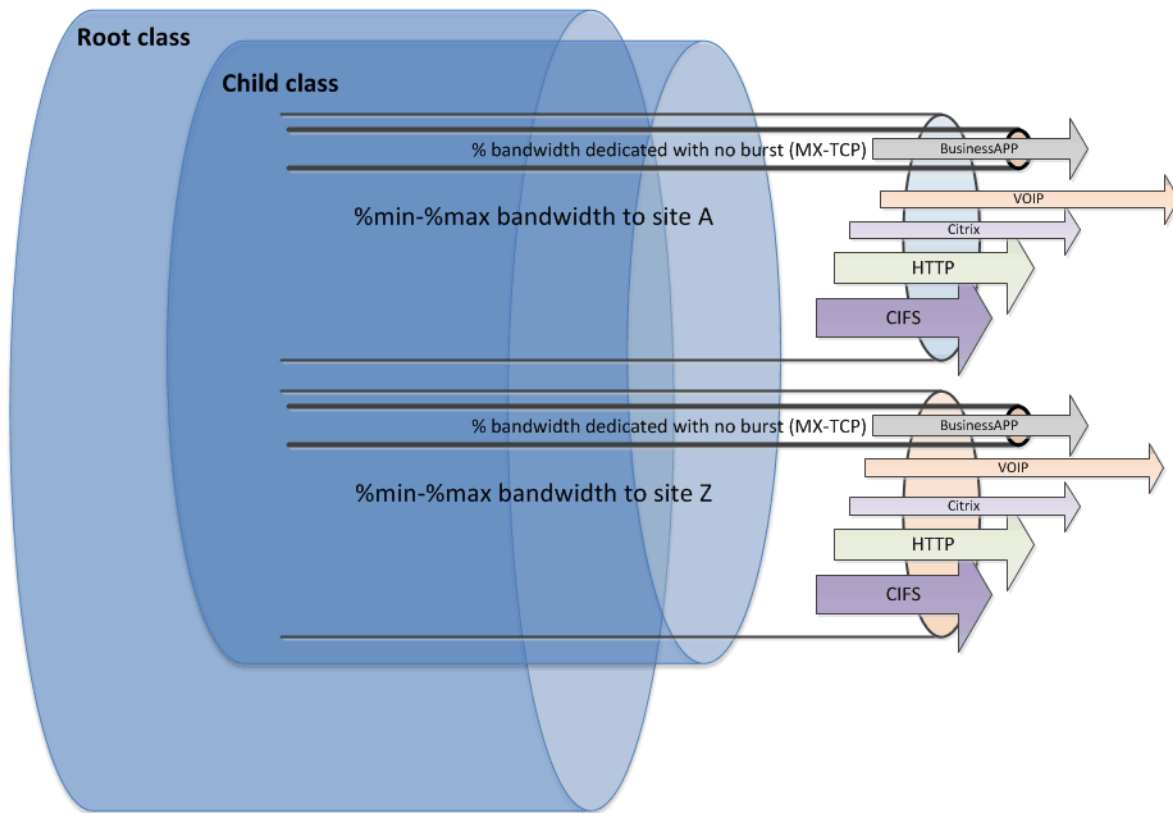


6. Draw the remaining remote site (Site B).

Replicate the same container as the first remote site (Site A). Remote sites tend to have the same configuration.

Your drawing should now look something like [Figure 5-3](#).

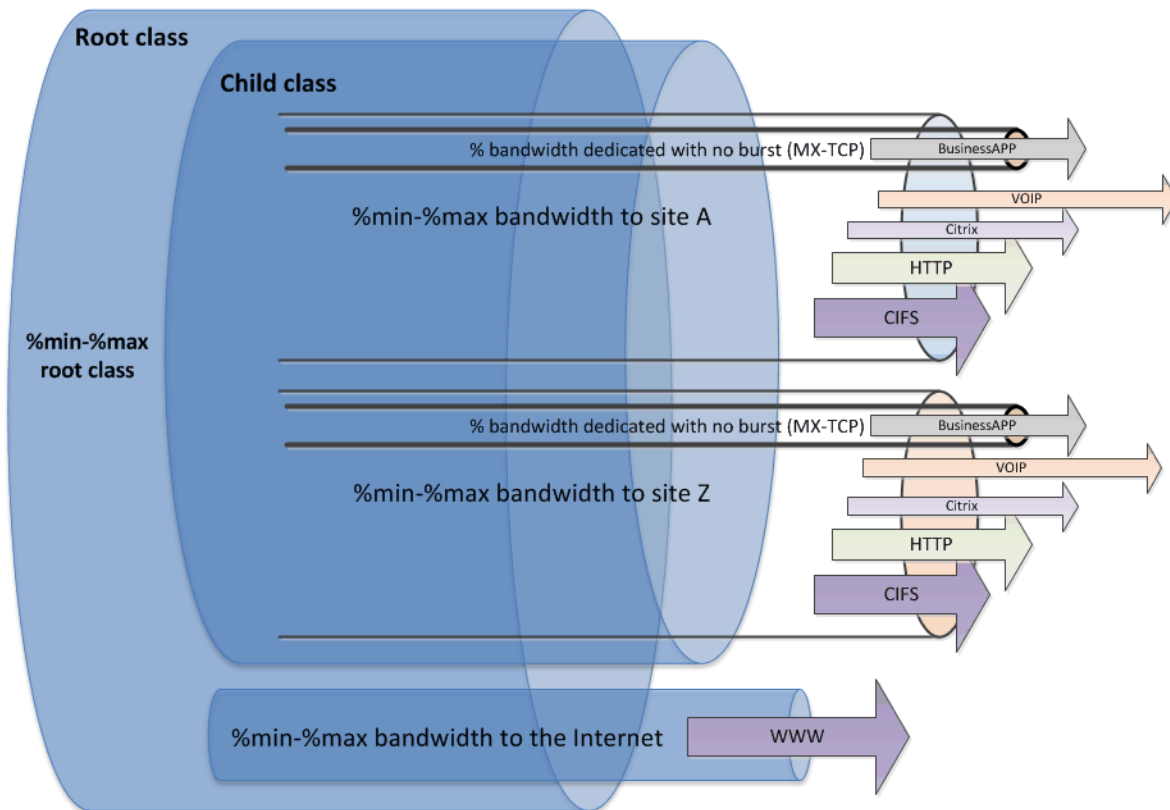
Figure 5-3. Draw the Remaining Remote Site



You can continue building containers nested inside other containers until you feel that you have defined all the traffic on which you want to apply QoS, assuming that you keep within the Steelhead appliance site and class limits.

For details on site and class limits, see [“Guidelines for the Maximum Number of QoS Classes, Sites, and Rules” on page 100.](#)

Figure 5-4. Adding Another Container



Keep in mind that there is always a default class associated with a site, so if traffic does not match an existing rule, it operates according to the default rule.

Configuring QoS Using Best Practices

This section describes an example network and the basic steps for configuring Riverbed QoS using the given specifications. This section includes the following topics:

- [“Example QoS Scenario” on page 113](#)
- [“Configuring QoS on the Data Center Steelhead Appliance” on page 115](#)
- [“Configuring QoS on the Branch Office Steelhead Appliance” on page 119](#)

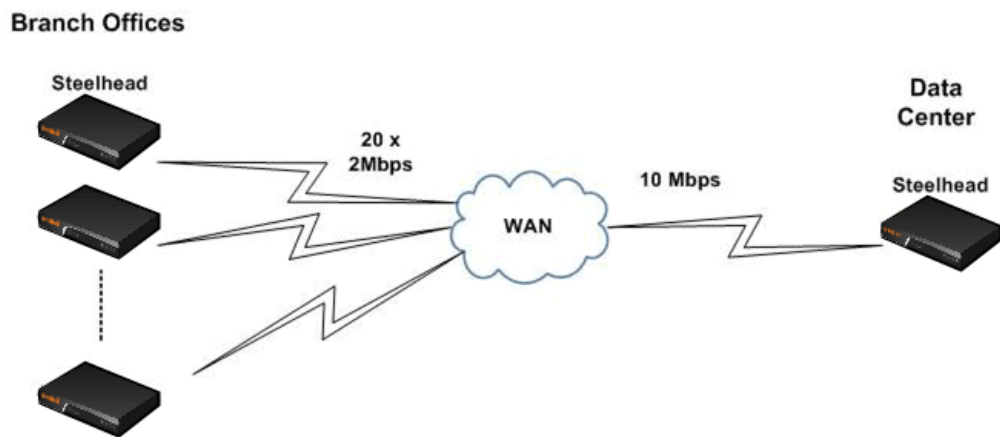
For details on best practices, see [“QoS Enforcement Best Practices” on page 105.](#)

Example QoS Scenario

This scenario is the basis for the configuration shown in [“Configuring QoS on the Data Center Steelhead Appliance” on page 115](#) and [“Configuring QoS on the Branch Office Steelhead Appliance” on page 119.](#)

In this example, traffic between the data center and the remote office branches includes VoIP, Citrix, software updates, and other traffic.

Figure 5-5. Steelhead Appliance Configuration Example



The data center:

- has Citrix servers located in the 10.1.0.0/24 subnet.
- transmits software updates from a server with IP address 10.1.1.100.
- uses Steelhead appliances that:
 - are deployed physically in-path.
 - have a WAN link with 10 Mbps of bandwidth.
 - serve 20 remote branch offices.
- uses Riverbed QoS hierarchical mode.
- has the following QoS policies for outbound traffic:
 - For each site, VoIP traffic is guaranteed at least 100 Kbps when active.
 - For each site, Citrix traffic is guaranteed at least 100 Kbps when active.
 - VoIP traffic is guaranteed the highest latency priority, and Citrix gets the second highest.
 - Software updates are allocated the lowest latency priority.

Each branch office has:

- a 2 Mbps WAN link.
- Steelhead appliances that are deployed physically in-path.
- a separate 10.16.X.0/24 subnet, where X is the number of the site.
- VoIP phones that are always in the 10.16.X.128/25 subnet.
- Riverbed QoS flat mode enabled.

Configuring QoS on the Data Center Steelhead Appliance

This section describes how to configure QoS on the data center Steelhead appliance. The first configuration shows the overview and the following sections describe each step in detail:

- [“Calculating the Effective Guaranteed Minimum Bandwidth” on page 115](#)
- [“Configuring the Data Center Service Policies” on page 116](#)
- [“Creating New Applications” on page 116](#)
- [“Configuring the Data Center with the List of Remote Sites” on page 118](#)

To configure Riverbed QoS on the data center Steelhead appliance

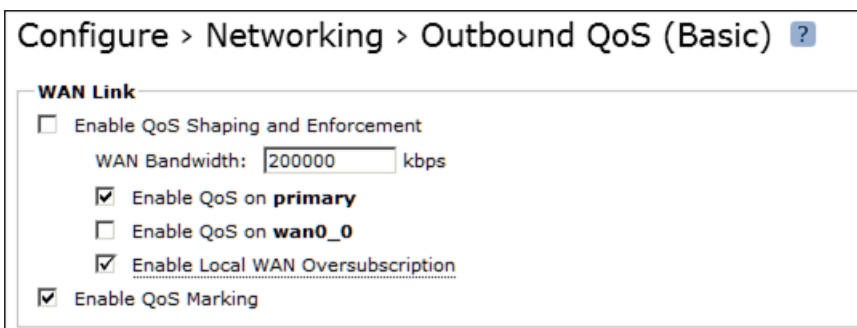
1. Enable WAN oversubscription and calculate the effective guaranteed bandwidth per site.
2. Create a new service policy.
3. Add new applications in the global application list.
4. Add remote sites to the list using the same service policy.
5. Configure the data center site-based classes.

Calculating the Effective Guaranteed Minimum Bandwidth

Consider an environment in which there are 20 remote sites, each with 2-Mbps link. The scale factor is 0.25 (10-Mbps physical link speed divided by the aggregate bandwidth of 40 Mbps from all the remote sites). This means that the effective guaranteed minimum bandwidth is approximately 500 Kbps per site. In other words, if all sites are running at the same time, each site is guaranteed at least 500 Kbps of bandwidth.

You must enable WAN oversubscription, as shown in [Figure 5-6](#).

Figure 5-6. Enable WAN Oversubscription



Configure > Networking > Outbound QoS (Basic) ?

WAN Link

- ☐ Enable QoS Shaping and Enforcement
 - WAN Bandwidth: kbps
- ☒ Enable QoS on **primary**
- ☐ Enable QoS on **wan0_0**
- ☒ Enable Local WAN Oversubscription
- ☒ Enable QoS Marking

Configuring the Data Center Service Policies

Voice traffic must have Realtime latency, although the Citrix traffic has the latency priority of Interactive. Configure each class to have a guaranteed minimum bandwidth of at least 100 Kbps.

Figure 5-7. Data Center Service Policies

Shaping Parameters		
	Minimum Bandwidth %	Maximum Bandwidth %
Realtime:	20	100
Interactive:	20	100
Business-Critical:	1	100
Normal:	49	100
Low-Priority:	9	100
Best-Effort:	1	100

100% total guaranteed

Assign 20% as guaranteed bandwidth for Realtime and Interactive latencies (100 Kbps/500 Kbps is 20%). Because the Business-Critical and Best-Effort classes are not used, assign them a bandwidth guarantee of 1%. For the Low-Priority class, use a low bandwidth guarantee. This ensures that when the Software Updates class and the Site Default class are both active, the software updates class receives a small percentage of any excess bandwidth over the minimum allocated to both QoS classes.

In [Figure 5-7](#), all classes have been set to 100% so that they can all use the full 2-Mbps pipe if the bandwidth is available.

Creating New Applications

Many commonly used applications are listed in the global application list. For example, H.323 and SIP voice applications are in the default list of applications and they already belong to the correct class. You do not need to modify these applications.

Figure 5-8. Common Applications

<input type="checkbox"/>	71	H.323	Realtime	h323
<input type="checkbox"/>	72	RTCP	Interactive	rtcp
<input type="checkbox"/>	73	RTP	Realtime	rtp
<input type="checkbox"/>	74	SIP	Realtime	sip

Citrix has an entry in the global application list. However, the default class associated with Citrix is Realtime, although the requirement for Citrix is the Interactive class. Although you can modify the default list, Riverbed does not recommend this, because doing so affects all sites. Riverbed recommends that you create a new application, as shown in [Figure 5-9](#).

Figure 5-9. Creating a New Citrix Application

The screenshot shows the 'Add Application' dialog box in the Riverbed Steelhead configuration interface. The dialog has three buttons at the top: 'Add Application' (selected), 'Remove Application', and 'Move Application'. The main form contains the following fields and settings:

- Name:** CU-Citrix-ICA
- Description:** Citrix - Interactive Service Class
- Position:** Start
- For Traffic with the Following Characteristics:**
 - Local Subnet:** 10.1.0.0/24
 - Remote Subnet:** 0.0.0.0/0
 - Protocol:** All
 - VLAN Tag ID:** all
 - DSCP:** All
 - Traffic Type:** All
 - Application:** ICA
- Apply these QoS Settings:**
 - Service Class:** Interactive
 - DSCP:** Inherit from Service Class

An 'Add' button is located at the bottom left of the form.

You must create a new application and assign the traffic to the default class. The application rule should appear at the very bottom of the list.

Figure 5-10. Default Application to Match the Rest of the Traffic in the Normal Class

▼ Add Application — Remove Application ⇅ Move Application

Name:

Description:

Position:

For Traffic with the Following Characteristics:

Local Subnet: Port:

Remote Subnet: Port:

Protocol:

VLAN Tag ID:

DSCP:

Traffic Type:

Application:

Apply these QoS Settings:

Service Class:

DSCP:

Configuring the Data Center with the List of Remote Sites

By default, the bandwidth associated with the Default-Site value is automatically set to the same speed as the WAN link speed. The Default-Site cannot be removed, but if you do not use it, then you can adjust the WAN throughput to minimize the influence it has when calculating the scaling factor for WAN oversubscription. In this example, the Default-Site is set to 5 Kbps.

To configure the data center with remote sites

1. Add one site by specifying the subnet, WAN throughput, and service policy.

2. Repeat the same process for the rest of the sites.

Figure 5-11. Adding Remote Sites

3. Verify the QoS configuration on the Reports > Appliance > QoS Statistics Dropped page and the Reports > Appliance > QoS Statistics Sent page.

Configuring QoS on the Branch Office Steelhead Appliance

This section describes how to configure QoS on the data center Steelhead appliance. The first configuration shows the overview and the following sections describe each step in detail:

- [“Creating a New Service Policy” on page 119](#)
- [“Adding New Applications to the Applications List” on page 120](#)
- [“Modifying the Service Policy for the Default-Site” on page 120](#)

You can use the basic outbound QoS mode with the branch office Steelhead appliances. Because the branch office Steelhead appliance only communicates with the data center, you do not need to enable WAN oversubscription.

To configure Riverbed QoS for the branch office Steelhead appliance

1. Enable QoS and set the WAN throughput to 2 Mbps.
2. Create a new service policy.
3. Add new applications to the global application list.
4. Add remote sites to the list using the same service policy.

Creating a New Service Policy

You must enable basic outbound QoS and set the bandwidth to 2 Mbps to create a new service policy. Configure the service policy with the following settings:

- Realtime: 5% (100 Kbps/2000 Kbps)
- Interactive: 5% (100 Kbps/200 Kbps)
- Business-Critical: 0.1%
- Normal: 13%

- Low-Priority: 2%
- Best-Effort: 0.1%

Figure 5-12. Branch Office Bandwidth Settings

	Minimum Bandwidth %	Maximum Bandwidth %
Realtime:	5	100
Interactive:	5	100
Business-Critical:	0.1	100
Normal:	13	100
Low-Priority:	2	100
Best-Effort:	0.1	100

25.200000000000003% total guaranteed

Add

Adding New Applications to the Applications List

Add the new applications to the application list by swapping the source and destination subnet where necessary. Create the default class.

For more details on adding new applications to the application list, see the *Steelhead Appliance Management Console User's Guide*.

Modifying the Service Policy for the Default-Site

Because the branch office Steelhead appliance communicates only with the data center, you can change the service policy associated with the Default-Site instead of creating a new site, as shown in [Figure 5-13](#).

Figure 5-13. Modifying the Service Policy for the Default-Site

Order ↑↓	Site Name ↑↓	Subnet ↑↓	Service Policy ↑↓	Remote Link Bandwidth ↑↓
1	☒ Default-Site	0.0.0.0/0	Medium_Office	20.00 Mbps

Subnet: 0.0.0.0/0

Remote Link Bandwidth: 20000 kbps

Service Policy: Data_Center

Apply

Configuring QoS Marking on Steelhead Appliances

You can mark incoming traffic to a LAN port of a Steelhead appliance with a DSCP or an IP TOS value (for details on default behavior, see [“QoS Marking Default Setting” on page 81](#)). In RiOS v7.0 or later, the DSCP or IP TOS marking has only local significance. This means that you can set the DSCP or IP TOS value on the server-side Steelhead appliance to a value that is different from those set on the client-side Steelhead appliance.

For more details on QoS marking settings, see the *Steelhead Appliance Management Console User’s Guide*.

Note: Prior to RiOS v7.0, the DSCP or IP TOS value on a server-side Steelhead appliance was determined by the DSCP or IP TOS value of the client-side Steelhead appliance. If you are running an earlier version of RiOS, see an earlier version of this guide for instructions on how to configure the DSCP or IP TOS value.

To enable QoS marking on a Steelhead appliance

1. Choose Configure > Networking > Outbound QoS (Basic).
2. Select Enable QoS Marking.

Figure 5-14. Enable QoS Marking

Configure > Networking > Outbound QoS (Basic) ?

QoS Settings

- ☐ Enable QoS Shaping
 - WAN Bandwidth: kbps
 - ☐ Enable QoS on **primary**
 - ☒ Enable QoS on **wan0_0**
 - ☐ Enable QoS on **wan0_1**
 - ☐ Enable Local WAN Oversubscription
- ☒ Enable QoS Marking
 - Global DSCP:

You can set the following traffic characteristics as the DSCP or IP TOS value:

- Source and destination IP address
- TCP/UDP port number
- Protocol or protocol number
- VLAN-ID
- Incoming DSCP or IP TOS value
- Traffic type (optimized, unoptimized, or all)
- Application (based on AFE)

You can also set a per-service policy in basic outbound QoS mode or QoS class in advanced outbound QoS mode.

To set the DSCP or IP TOS value based on traffic characteristics in basic outbound QoS mode

1. Choose Configure > Networking > Outbound QoS (Basic).
2. Select the Applications tab.

Edit an existing application or select Add Application to add a new application.

Figure 5-15. Setting DSCP or IP TOS Value Based on Traffic Characteristics in Basic Outbound QoS Mode

The screenshot shows the 'Applications' tab in the configuration interface. At the top are tabs for 'Sites', 'Applications', and 'Service Policies'. Below the tabs are three buttons: 'Add Application' (with a dropdown arrow), 'Remove Application', and 'Move Application' (with up/down arrows). The main form has the following fields:

- Name:** A text input field with a red dot icon.
- Description:** A text input field.
- Position:** A dropdown menu with 'End' selected.
- For Traffic with the Following Characteristics:**
 - Source Subnet:** Text input '0.0.0.0/0'.
 - Destination Subnet:** Text input '0.0.0.0/0'.
 - Port:** Two dropdown menus, both set to 'all'.
 - Protocol:** Dropdown menu set to 'All'.
 - VLAN Tag ID:** Text input 'all'.
 - DSCP:** Text input 'all'.
 - Traffic Type:** Dropdown menu set to 'All'.
 - Application:** Dropdown menu set to '--'.
- Apply these QoS Settings:**
 - Service Class:** Dropdown menu set to 'Realtime'.
 - DSCP:** Dropdown menu set to 'Inherit from Service Class'.

An 'Add' button is located at the bottom left of the form.

To set DSCP or IP TOS value per service policy in basic outbound QoS

1. Choose Configure > Networking > Outbound QoS (Basic).
2. Select the Service Policies tab.

Edit an existing service policy or select Service Policies to add a new service policy.

Figure 5-16. Setting DSCP or IP TOS Value Based on Service Policy in Outbound QoS (Basic) Mode

Sites Applications **Service Policies**

▼ Add Service Policy — Remove Service Policy

Name:

	Shaping Parameters		Marking Parameters
	Minimum Bandwidth %	Maximum Bandwidth %	DSCP
Realtime:	<input type="text" value="0"/>	<input type="text" value="100"/>	<input type="text" value="Reflect"/>
Interactive:	<input type="text" value="0"/>	<input type="text" value="100"/>	<input type="text" value="Reflect"/>
Business-Critical:	<input type="text" value="0"/>	<input type="text" value="100"/>	<input type="text" value="Reflect"/>
Normal:	<input type="text" value="0"/>	<input type="text" value="100"/>	<input type="text" value="Reflect"/>
Low-Priority:	<input type="text" value="0"/>	<input type="text" value="100"/>	<input type="text" value="Reflect"/>
Best-Effort:	<input type="text" value="0"/>	<input type="text" value="100"/>	<input type="text" value="Reflect"/>

0% total guaranteed

Add

To set DSCP or IP TOS value based on traffic characteristics in advanced outbound QoS mode

1. Choose Configure > Networking > Outbound QoS (Advanced).
2. Scroll down to Sites and Rules.

Edit an existing rule or select Add a Site or Rule to add a site or rule.

Figure 5-17. Setting DSCP or IP TOS Value Based on Traffic Characteristic in Advanced Outbound QoS Mode

QoS Sites and Rules:

▼ Add a Site or Rule — Remove Sites or Rules ↑↓ Move...

Add a: ☐ Site ☒ Rule

Name:

Description:

Parent Site:

Insert Rule At:

For Traffic with the Following Characteristics:

Local Subnet or [Host Label](#): Port or [Port Label](#):

Remote Subnet or [Host Label](#): Port or [Port Label](#):

Protocol:

VLAN Tag ID:

DSCP:

Traffic Type:

Application:

Apply these QoS Settings:

Service Class:

DSCP:

Apply these Path Selections:

[Path](#) preference order (only one path will be used):

Path 1: DSCP:

Path 2: DSCP:

Path 3: DSCP:

If no configured path is up:

☒ Relay traffic from the interface normally.

☐ Drop traffic.

To set DSCP or IP TOS value per QoS class in advanced outbound QoS mode

1. Choose Configure > Networking > Outbound QoS (Advanced).
2. Scroll to the QoS Class section.

Edit an existing QoS class or select Add a New Class to add a new QoS class.

Figure 5-18. Setting DSCP or IP TOS Value Based on QoS Class in Advanced Outbound QoS Mode

QoS Classes:

▼ Add a New Class — Remove Selected

Name:

Shaping Parameters:

Class Parent:

Queue:

Minimum Bandwidth: %

Maximum Bandwidth: %

Latency Priority:

Connection Limit:

Marking Parameters:

DSCP:

Add

The DSCP or IP TOS value that is defined in the application rules (basic outbound QoS) or in the rules configuration (advanced outbound QoS) take precedence over the values defined in the service class (basic outbound QoS) or in the QoS class (advanced outbound QoS).

Riverbed recommends that you do the following:

- For QoS marking only, set the DSCP or IP TOS value in the application rules (basic outbound QoS) or in the rules configuration (advanced outbound QoS).
- For QoS marking and traffic shaping, set the DSCP or IP TOS value in the service class (basic outbound QoS) or in the QoS class (advanced outbound QoS).

Configuring QoS for Citrix Traffic

This section contains the following topics:

- [“Configuring Basic Outbound QoS and Citrix Traffic in a Pure Interactive Environment” on page 125](#)
- [“Configuring Inbound QoS and Citrix Traffic” on page 127](#)
- [“Configuring Advanced Outbound QoS and Citrix Traffic in a Mixed-Traffic Environment” on page 131](#)

For more information about Citrix, see the *Steelhead Appliance Deployment Guide - Protocols*.

Configuring Basic Outbound QoS and Citrix Traffic in a Pure Interactive Environment

This section describes how to configure basic outbound QoS to identify Citrix traffic in a pure interactive environment.

To configure basic outbound QoS to identify Citrix traffic in a pure interactive environment

1. Confirm the correct class setting for the Citrix ICA client.
 - Choose Configure > Networking > Outbound QoS (Basic).
 - Select the Applications tab.

Figure 5-19. Outbound QoS (Basic) Page, Applications Tab

Configure > Networking > Outbound QoS (Basic) ?

QoS Settings

☒ Enable QoS Shaping

WAN Bandwidth: kbps

☐ Enable QoS on **primary**

☒ Enable QoS on **wan0_0**

☐ Enable Local WAN Oversubscription

☐ Enable QoS Marking

Global DSCP:

Apply

Sites Applications Service Policies

+ Add Application - Remove Application ↑↓ Move Application

<input type="checkbox"/>	Order ↑↓	Name ↑↓	Service Class ↑↓	DSCP ↑↓
<input type="checkbox"/>	1	ICMP	Business-Critical	from Class
<input type="checkbox"/>	2	ICA	Business-Critical	from Class
<input type="checkbox"/>	3	CIFS	Normal	from Class
<input type="checkbox"/>	4	NFS	Normal	from Class
<input type="checkbox"/>	5	iTunes	Low-Priority	from Class
<input type="checkbox"/>	6	RDP	Business-Critical	from Class
<input type="checkbox"/>	7	DNS	Business-Critical	from Class
<input type="checkbox"/>	8	Facebook	Low-Priority	from Class

- Confirm that the Citrix ICA client is in the correct class.
2. Confirm the correct service policies.
 - Choose Configure > Networking > Outbound QoS (Basic).

- Select the Service Policies tab.

Figure 5-20. Outbound QoS (Basic) Page, Service Policies

Sites Applications Service Policies							
+ Add Service Policy – Remove Service Policy							
<input type="checkbox"/>	Name ↑↓	Realtime	Interactive	Business-Critical	Normal	Low-Priority	Best-Effort
<input type="checkbox"/>	Large_Office	10-100% Reflect	10-100% Reflect	20-100% Reflect	50-100% Reflect	9-100% Reflect	
<input type="checkbox"/>	Larger_Office	20-100% Reflect	20-100% Reflect	20-100% Reflect	20-100% Reflect	19-100% Reflect	
<input type="checkbox"/>	Medium_Office	10-100% Reflect	20-100% Reflect	20-100% Reflect	40-100% Reflect	9-100% Reflect	
<input type="checkbox"/>	Small_Office	20-100% Reflect	20-100% Reflect	30-100% Reflect	20-100% Reflect	9-100% Reflect	
<input type="checkbox"/>	Smaller_Office	1-100% Reflect	1-100% Reflect	40-100% Reflect	40-100% Reflect	17-100% Reflect	

- Confirm that the default service policies meet your requirements. If necessary, create a new service policy.

3. Add a new site (optional).

By default, the service policy Medium_Office is associated with the site named Default-Site. Default-Site is a catch-all site with a subnet of 0.0.0.0/0. Riverbed recommends that you create a new site.

Figure 5-21. Add Remote Site Tab

Sites Applications Service Policies					
+ Add Site – Remove Site ↑↓ Move Site					
<input type="checkbox"/>	Order ↑↓	Name ↑↓	Remote Subnets ↑↓	Service Policy ↑↓	Remote Bandwidth
<input type="checkbox"/>	1	London	10.22.1.0/24	Small_Office	2048.0 Kbps
<input type="checkbox"/>	2	Paris	10.22.2.0/24	Large_Office	5120.0 Kbps
<input type="checkbox"/>	3	Default-Site	0.0.0.0/0	Medium_Office	2000.0 Kbps

Figure 5-21 shows that Citrix traffic (and other traffic belong to the Business-Critical class) destined for the Paris subnet has a guaranteed bandwidth of 1024 Kbps (20% of 5120 Kbps for Business-Critical class). Business-Critical traffic destined to the London subnet has a guaranteed bandwidth of 614.4 Kbps (30% of 2048 Kbps). Business-Critical traffic for all other locations has a guaranteed bandwidth of 2000 Kbps.

Configuring Inbound QoS and Citrix Traffic

This section shows how to configure inbound QoS to properly classify Citrix traffic. The example shows configurations for normal Citrix (optimized and passthrough), as well as for Citrix over SSL.

The packet-order queue is not available for inbound QoS classification. Therefore, with inbound QoS you can apply classification on the entire Citrix stream as only a single flow and you cannot separate the different traffic priority levels into different classes as you can with outbound QoS. Inbound QoS configuration for Citrix is the same process for either optimized and passthrough traffic. You do not need to perform any additional configuration to properly classify and identify the different types of traffic.

To configure inbound QoS to identify normal Citrix traffic (commonly known as ICA or CGP)

1. Choose Configure > Networking > Inbound QoS.
2. Enable the desired interface.
3. Specify the WAN bandwidth for the interface.
4. Select Enable Inbound QoS Shaping and Enforcement.

Figure 5-22. Enable Inbound QoS Shaping and Enforcement

Configure > Networking > Inbound QoS ?

WAN Link

☒ Enable Inbound QoS Shaping and Enforcement

Network Interfaces:

☒ Enable QoS on **wan0_0** with WAN Bandwidth: 15000 kbps

☐ Enable QoS on **wan1_0** with WAN Bandwidth: kbps

Apply

Next, create the multiple classes for the various traffic for which you want to apply inbound QoS shaping and enforcement—this is not just reserved for Citrix. All traffic you want to shape and enforce as inbound QoS affects all traffic the Steelhead appliance is receiving.

To create a new class

1. Select Add a Class.
2. Select the desired priority from the drop-down list.
3. Specify the minimum and maximum bandwidth and optionally specify the link share weight.
4. Click Add.

Figure 5-23. Add a Class

Inbound QoS Classes:

▼ Add a Class — Remove Selected

Class Name:

Priority: Realtime

Minimum Bandwidth: 0 %

Maximum Bandwidth: 100 %

Link Share Weight: 100

Add

Figure 5-24 shows multiple classes configured along with a Citrix class associated in the interactive priority.

Figure 5-24. Multiple Configured Classes with a Citrix Class

Inbound QoS Classes:

[+ Add a Class](#) [- Remove Selected](#)

<input type="checkbox"/>	Name ↑↓	Priority ↑↓	Min BW % ↑↓	Max BW % ↑↓	Link Share Weight ↑↓
<input type="checkbox"/>	cifs	Normal	20.0	50.0	100.0
<input type="checkbox"/>	Citrix	Interactive	20.0	100.0	100.0
<input type="checkbox"/>	rdp	Interactive	15.0	100.0	100.0
<input type="checkbox"/>	Recreational	Best-Effort	2.0	100.0	100.0
<input type="checkbox"/>	VOIP	Realtime	33.0	100.0	100.0
<input type="checkbox"/>	youtube	Best-Effort	0.0	100.0	100.0
<input type="checkbox"/>	Default	Normal	10.0	100.0	100.0

Next, create the rules in which you can specify the proper application and associate it with the class you previously created.

To create an inbound QoS rule

1. Select Add a Rule.
2. Specify a name, and optionally specify a description.
3. Select the rule number precedence (inbound QoS is similar to outbound QoS rule sets in that it matches first logic).
4. Select and specify the characteristics.
For example, you can choose characteristics based on any header parameter, such as source and/or destination IP and/or port. Riverbed recommends that you use the AFE in the application parameter, and select the name of the desired application, such as Citrix.
5. Associate this rule with an already created class.
6. Click **Apply**.

Figure 5-25 shows a Citrix rule being configured with ICA as an application and Citrix as a service class.

Figure 5-25. Add a Rule

2 Citrix Citrix ICA

Name:

Description:

For Traffic with the Following Characteristics:

Remote Subnet or **Host Label**: Port or **Port Label**:

Local Subnet or **Host Label**: Port or **Port Label**:

Protocol:

Traffic Type:

DSCP:

VLAN Tag ID:

Application:

Apply these QoS Settings:

Service Class Name:

Figure 5-26 shows multiple rules created along with the Citrix rule. The example shows VoIP traffic as the highest order and priority, followed by Citrix. The example also shows the desire to limit recreational Internet-bound traffic to a lower priority.

Figure 5-26. Multiple Configured Rules including a Citrix Rule

Inbound QoS Rules:

<input type="checkbox"/>	Order	Name	Service Class	Application
<input type="checkbox"/>	1	VOIP	VOIP	Google-Talk
<input type="checkbox"/>	2	Citrix	Citrix	ICA
<input type="checkbox"/>	3	RDP	rdp	RDP
<input type="checkbox"/>	4	CIFS	cifs	CIFS
<input type="checkbox"/>	5	facebook	Recreational	Facebook
<input type="checkbox"/>	6	twitter	Recreational	Twitter
<input type="checkbox"/>	7	skype	Recreational	Skype
<input type="checkbox"/>	8	youtube	Recreational	Youtube
<input type="checkbox"/>	9	dropbox	Recreational	Dropbox
	default	default	Default	

To apply inbound QoS for Citrix over SSL traffic, you must create a specific rule to properly identify the traffic. If the data stream is optimized, then the normal ICA application is able to properly identify the traffic (as detailed in the example).

To configure Citrix over SSL in passthrough mode

1. Create a new rule (in addition to the Citrix rule that classifies optimized Citrix over SSL traffic and traditional ICA and CGP traffic) and specify the proper source IP and/or source port of the remote server.

This forces classification to trigger on those metrics and you can reuse the Citrix class you already created for the previous Citrix class.

Figure 5-27. Citrix Over SSL

Inbound QoS Rules:

[+ Add a Rule](#) [- Remove Rules](#) [↑↓ Move...](#)

<input type="checkbox"/>	Order	Name	Service Class	Application
<input type="checkbox"/>	1	VOIP	VOIP	Google-Talk
<input type="checkbox"/>	2	Citrix	Citrix	ICA
<input type="checkbox"/>	3	CitrixOverSSL	Citrix	

Description: Passthrough Citrix over SSL

Name:

Description:

For Traffic with the Following Characteristics:

Remote Subnet or [Host Label](#): Port or [Port Label](#):

Local Subnet or [Host Label](#): Port or [Port Label](#):

Protocol:

Traffic Type:

DSCP:

VLAN Tag ID:

Application:

Apply these QoS Settings:

Service Class Name:

[Apply](#)

<input type="checkbox"/>	4	RDP	rdp	RDP
<input type="checkbox"/>	5	CIFS	cifs	CIFS
<input type="checkbox"/>	6	facebook	Recreational	Facebook
<input type="checkbox"/>	7	twitter	Recreational	Twitter
<input type="checkbox"/>	8	skype	Recreational	Skype
<input type="checkbox"/>	9	youtube	Recreational	Youtube
<input type="checkbox"/>	10	dropbox	Recreational	Dropbox
	default	default	Default	

Configuring Advanced Outbound QoS and Citrix Traffic in a Mixed-Traffic Environment

This section shows how to configure advanced outbound QoS to identify Citrix traffic in a mixed-traffic environment. You must create three QoS classes:

- Interactive traffic
- Normal, non-interactive traffic
- Default class to catch any Citrix traffic that cannot be classified

Use the AFE to create a single rule and associate the different classes with the various application priority.

To configure advanced outbound QoS to identify Citrix traffic in a mixed-traffic environment**1. Define an interactive QoS class for Citrix:**

- Choose Configure > Networking > Outbound QoS (Advanced).
- Under QoS Classes, select Add a New QoS Class.
- Specify the class name: for example, `CitrixInteractive`.
- From the Latency Priority drop-down list, select Real-Time.
- Specify a minimum bandwidth that is appropriate for your environment.
- From the Queue drop-down list, select packet-order.
- Click **Add**.

Figure 5-28. Define a Citrix QoS Class for Interactive Traffic

The screenshot shows a window titled "QoS Classes:" with two buttons at the top: "▼ Add a New Class" and "— Remove Selected". Below these buttons, the "Name:" field contains "CitrixInteractive". Under the "Shaping Parameters:" section, the "Class Parent:" is set to "root", "Latency Priority:" is set to "Realtime", "Minimum Bandwidth:" is "20 %", "Maximum Bandwidth:" is "100 %", "Optimized Connection Limit:" is an empty field, and "Queue:" is set to "packet-order". Under the "Marking Parameters:" section, the "DSCP:" is set to "Reflect". An "Add" button is located at the bottom left of the dialog.

2. Define a non-interactive QoS class for Citrix:

- Choose Configure > Networking > Outbound QoS (Advanced).
- Under QoS Classes, select Add a New QoS Class.
- Specify the class name: for example, `CitrixNormal`.
- From the Latency Priority drop-down list, select Normal Priority.
- Specify a minimum bandwidth that is appropriate for your environment.
- From the Queue drop-down list, select packet-order.

- Click **Add**.

Figure 5-29. Defining a Citrix QoS Class for Normal Traffic

QoS Classes:

▼ Add a New Class — Remove Selected

Name:

Shaping Parameters:

Class Parent:

Latency Priority:

Minimum Bandwidth: %

Maximum Bandwidth: %

Optimized Connection Limit:

Queue:

Marking Parameters:

DSCP:

- Define a default QoS class for Citrix:
 - Choose **Configure > Networking > Outbound QoS (Advanced)**.
 - Under **QoS Classes**, select **Add a New QoS Class**.
 - Specify the class name: for example, `CitrixDefault`.
 - From the **Latency Priority** drop-down list, select **Low Priority**.
 - Specify a minimum bandwidth that is appropriate for your environment.
 - From the **Queue** drop-down list, select **packet-order**.
 - Click **Add**.
- When you are finished, confirm the QoS classes, as shown in [Figure 5-30](#).

Figure 5-30. Citrix QoS Classes

QoS Classes:							
+ Add a New Class — Remove Selected							
<input type="checkbox"/>	Name	Latency Priority	Min BW %	Max BW %	Conn Limit	Queue	DSCP
<input type="checkbox"/>	🔍 CitrixDefault	Low-Priority	20.00	100.00		packet-order	Reflect
<input type="checkbox"/>	🔍 CitrixInteractive	Interactive	20.00	100.00		packet-order	Reflect
<input type="checkbox"/>	🔍 CitrixNormal	Normal	40.00	100.00		packet-order	Reflect
▶	🔍 Default-Site\$parent_class	Normal	10.00	10.00		sfq	Reflect

- Define QoS rules that assign an application priority to certain types of traffic.

This directs traffic to the classes, ensuring that all Citrix traffic is diverted to one of the packet-ordered classes. Define the Citrix QoS rules on both the client-side and server-side Steelhead appliances.

To use session reliability (port number 2598), you must enable Citrix optimization on the Steelhead appliance to classify the traffic correctly. You can enable and modify Citrix ICA optimization settings in the **Configure > Optimization > Citrix ICA** page. For details, see the *Steelhead Appliance Management Console User's Guide*.

Use the following steps to define QoS rules for Citrix traffic:

- Choose Configure > Networking > Outbound QoS (Advanced).
- Under Add a QoS Site or Rules, select Add a rule.
- From the Class Name drop-down list, select CitrixDefault.
- From the Application drop-down list, select ICA.
- Select the desired class for the different application priority.
- If necessary, change the source and destination subnet.
- Click Add.

Figure 5-31. Define QoS Rules for Citrix Traffic

QoS Sites and Rules:

▼ Add a Site or Rule — Remove Sites or Rules ⇅ Move...

Add a: ☐ Site ☒ Rule

Name: Rules for Citrix Traffic

Description:

Parent Site: Default-Site ▾

Insert Rule At: Start ▾

For Traffic with the Following Characteristics:

Local Subnet or [Host Label](#): 0.0.0.0/0 Port or [Port Label](#): all

Remote Subnet or [Host Label](#): 0.0.0.0/0 Port or [Port Label](#): all

Protocol: All ▾

VLAN Tag ID: all

DSCP: All ▾

Traffic Type: All ▾

Application: ICA

Apply these QoS Settings:

Service Class: CitrixDefault ▾ (for non-prioritized traffic)

DSCP: Inherit from Service Class ▾

	Service Class	DSCP
ICA Priority 0:	CitrixInteractive ▾	Inherit from Service Class ▾
ICA Priority 1:	CitrixInteractive ▾	Inherit from Service Class ▾
ICA Priority 2:	CitrixNormal ▾	Inherit from Service Class ▾
ICA Priority 3:	CitrixNormal ▾	Inherit from Service Class ▾

When the main Service Class has a packet-order queue type, the ICA Priority Service Classes can be any packet-order class.

- Confirm that the QoS rules appear as shown in [Figure 5-32](#).

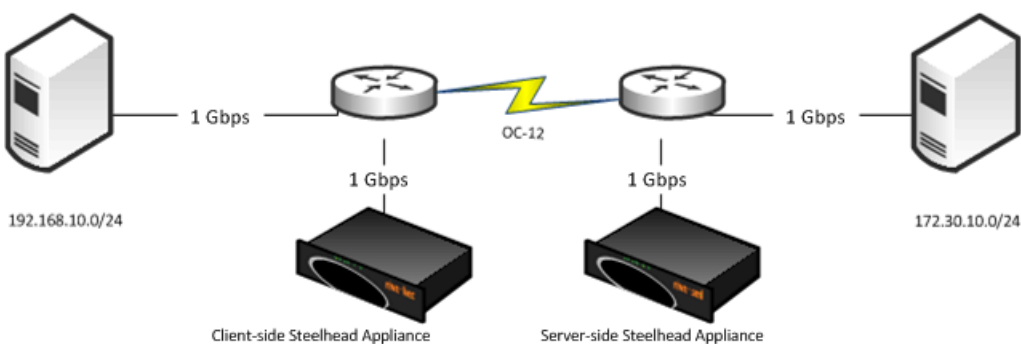
Figure 5-32. QoS Rules

QoS Sites and Rules:			
<input type="button" value="+ Add a Site or Rule"/> <input type="button" value="- Remove Sites or Rules"/> <input type="button" value="⇅ Move..."/>			
<input type="checkbox"/>	Order	Name	Service Class
		▼ Default-Site	
<input type="checkbox"/>	1	Rules for Citrix Traffic	Default-Site\$\$Best-Effort
	default	default	Default-Site\$\$Low-Priority

Configuring QoS and MX-TCP

This section described how to configure Riverbed QoS and MX-TCP on the client-side and the server-side Steelhead appliances.

Figure 5-33. MX-TCP Example



To configure QoS and MX-TCP on the client-side Steelhead appliance

- On the client-side Steelhead appliance, select Enable QoS Classification and Enforcement.
- Select Hierarchical.
- Enable QoS on the WAN interface and specify the WAN throughput to 1,000,000 Kbps (1 Gbps).

Set this to be the lower of LAN plus WAN or the speed of the interface. In [Figure 5-34](#), the LAN speed (between the server in the data center and the router) is 1 Gbps and the WAN speed is 622 Mbps. The combined speed is 1.622 Gbps. However, the link speed for the WAN interface on the Steelhead appliance is only 1 Gbps; therefore, the WAN link speed is set to 1 Gbps and not 1.622 Gbps.

Figure 5-34. Outbound QoS (Advanced) Page

Configure > Networking > Outbound QoS (Advanced) ?

QoS Settings

☒ Enable QoS Shaping

Mode (changing modes while QoS is enabled can cause momentary network disruptions):

☐ Flat ☒ Hierarchical

Network Interfaces:

☐ Enable QoS on **primary** with WAN Bandwidth: kbps

☒ Enable QoS on **wan0_0** with WAN Bandwidth: kbps

☐ Enable QoS Marking

Global DSCP:

During the conversion from basic outbound QoS to advanced outbound QoS, a default QoS class known as *Default-Site\$\$parent_class* is created. This class is given a minimum bandwidth of 100%. The minimum guaranteed bandwidth can never exceed a combined total of 100% and hence, in the scenario shown in [Figure 5-35](#), the link is dedicated for replication. The Riverbed recommended value for minimum bandwidth is 1%.

Figure 5-35. Default-Site\$\$parent Minimum Bandwidth

QoS Classes:

Name
▼ <input checked="" type="checkbox"/> Default-Site\$\$parent_class

Shaping Parameters:

Latency Priority:

Minimum Bandwidth: %

Maximum Bandwidth: %

Optimized Connection Limit:

Queue:

Marking Parameters:

DSCP:

4. Create a LAN class and a WAN class and note the different minimum bandwidth settings.

Figure 5-36. LAN Class Minimum Bandwidth

QoS Classes:
▼ Add a New Class — Remove Selected

Name:

Shaping Parameters:
Class Parent:
Latency Priority:
Minimum Bandwidth: %
Maximum Bandwidth: %
Optimized Connection Limit:
Queue:

Marking Parameters:
DSCP:

Figure 5-37. WAN Class Minimum Bandwidth

QoS Classes:
▼ Add a New Class — Remove Selected

Name:

Shaping Parameters:
Class Parent:
Latency Priority:
Minimum Bandwidth: %
Maximum Bandwidth: %
Optimized Connection Limit:
Queue:

Marking Parameters:
DSCP:

5. Create an MX-TCP class under the WAN class and assign 99% of the minimum bandwidth to this class.

Figure 5-38. MX-TCP Minimum Bandwidth

The screenshot shows a configuration window titled "QoS Classes:". At the top, there are two buttons: "▼ Add a New Class" and "— Remove Selected". Below these, the "Name:" field is set to "MXTCP". Under the "Shaping Parameters:" section, "Class Parent:" is set to "WAN Class", "Latency Priority:" is set to "Realtime", "Minimum Bandwidth:" is set to "60 %", "Maximum Bandwidth:" is set to "99 %", "Optimized Connection Limit:" is empty, and "Queue:" is set to "mxtcp". Under the "Marking Parameters:" section, "DSCP:" is set to "Reflect". At the bottom left, there is an "Add" button.

If the Steelhead appliance has a bandwidth limit, then the minimum bandwidth must be within it. The MX-TCP class can use up to 99% of the bandwidth of the WAN class. For details on configuring MX-TCP before RiOS v8.0, see earlier versions of the *Steelhead Appliance Deployment Guide* and *Riverbed Deployment Guide* on the Riverbed Support site at <https://support.riverbed.com>.

6. Add a new site (optional).

On the QoS Sites and Rules tab, there is a default site called Default-Site. Default-Site is a catch-all site with a subnet of 0.0.0.0/0. Riverbed recommends that you create a new site.

Figure 5-39. Create a New Site Instead of Default-Site

QoS Sites and Rules:

▼ Add a Site or Rule — Remove Sites or Rules ↑↓ Move...

Add a: ☒ Site ☐ Rule

Name:

Subnets:

Default Rule Settings:

Service Class:

DSCP:

Default Rule Path Selections:

Path preference order (only one path will be used):

Path 1: DSCP:

Path 2: DSCP:

Path 3: DSCP:

If no configured path is up:

☒ Relay traffic from the interface normally.

☐ Drop traffic.

7. Define a QoS rule to place the traffic from the data center (192.168.10.0/24) to the disaster recovery site (172.30.10.0/24) into the MX-TCP class.

- From the Traffic Type drop-down list, select Optimized and select the TCP protocol. Only optimized traffic is eligible to be placed in the MX-TCP class. Do not select any application, because the AFE is incompatible with MX-TCP.

Figure 5-40. QoS Rule for MX-TCP

QoS Sites and Rules:

▼ Add a Site or Rule — Remove Sites or Rules ⬆️⬆️ Move...

Add a: ☐ Site ☒ Rule

Name: Rule for MXTCP

Description:

Parent Site: DC-DR

Insert Rule At: End

For Traffic with the Following Characteristics:

Local Subnet or [Host Label](#): 192.168.10.0/24 Port or [Port Label](#): all

Remote Subnet or [Host Label](#): 172.20.10.0/24 Port or [Port Label](#): all

Protocol: TCP

VLAN Tag ID: all

DSCP: All

Traffic Type: Optimized

Application:

Apply these QoS Settings:

Service Class: MXTCP

DSCP: Inherit from Service Class

Apply these Path Selections:

[Path](#) preference order (only one path will be used):

Path 1: -- DSCP: Inherit DSCP from Rule

Path 2: -- DSCP: Inherit DSCP from Rule

Path 3: -- DSCP: Inherit DSCP from Rule

If no configured path is up:

☒ Relay traffic from the interface normally.

☐ Drop traffic.

Add

To configure QoS and MX-TCP on the server-side Steelhead appliance

Note: Configuring the server-side Steelhead appliance is similar to configuring the client-side Steelhead appliance.

- When you create the site, specify the IP address range of the disaster recovery site (172.30.10.0/24).

2. Create the QoS rule to place the traffic into the MX-TCP class, reversing the source and destination subnets.

Creating Host Labels

Use host labels to group multiple subnets or hostnames into one label. You create host labels on the Configure > Networking > Host Labels page.

Figure 5-41. Host Labels Page

Configure > Networking > Host Labels ?

Summary of Hostname Resolution

- 2 Unique Hostnames
- 2 Checking DNS
- 0 Unresolvable

Hostnames are automatically resolved once every day.

Resolve Hostnames

☐ Show resolved IPs for the hostnames in the table below

+ Add a New Host Label - Remove Selected

<input type="checkbox"/>	Label	Hostnames	Subnets
<input checked="" type="checkbox"/>	Exchange-Servers	exchgsrv03 <i>Checking DNS ...</i> exchgtest <i>Checking DNS ...</i>	10.84.18.0/24 10.84.19.9/32

Hostnames/Subnets:

exchgsrv03
 exchgtest
 10.84.18.0/24
 10.84.19.9/32

Entries can be separated with commas, spaces, or newlines.

Apply

After you create the host label, you can add the host label to a QoS rule to match a traffic source or a destination IP address. A host label matches a QoS rule with only a single common entry on the list of subnets or hostnames.

Figure 5-42. Adding the Host Label

▼ Add Application — Remove Application ⇅ Move Application

Name:

Description:

Position:

For Traffic with the Following Characteristics:

Local Subnet or **Host Label**: Port or **Port Label**:

Remote Subnet or **Host Label**: Port or **Port Label**:

Protocol:

VLAN Tag ID:

Use the following guidelines when creating a host label:

- If hostnames in a host label are not resolved by DNS, traffic does not match.
- You cannot use host labels to define sites.
- Host labels do not support IPv6.
- All configured hostnames are automatically resolved by DNS every 24 hours.
- Click **Resolve Hostnames** to immediately resolve hostnames through DNS.
- Any changes in IP addresses that a hostname resolves are relayed to the QoS rule.
- You can configure a maximum of 100 unique hostnames.
- There is a maximum of 64 subnets and hostnames per host label.
- There is no limit on the number of host labels that you can configure.

For more information about host labels, see the *Steelhead Appliance Management Console User's Guide*.

Configuring QoS for SSL Common Name Matching

QoS can use the AFE to classify incoming or outgoing SSL traffic based on the TLS/SSL server common name. After the traffic is classified, you can prioritize and shape it like all other configured applications.

For additional details, see [“Overview of Application Flow Engine” on page 83](#).

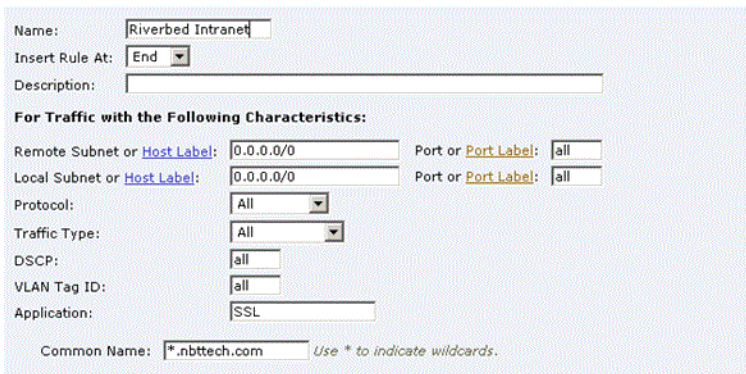
SSL common name matching is available in basic and advanced outbound QoS, and in inbound QoS.

To configure SSL common name matching in QoS (basic and advanced outbound, and inbound)

1. The first steps are different, depending on whether you are configuring on basic outbound QoS, basic advanced QoS, or inbound QoS.
 - For basic outbound QoS, choose Configure > Networking > Outbound QoS (Basic), select the Applications tab, and select Add Application.

- For advanced outbound QoS, choose Configure > Networking > Outbound QoS (Advanced), select Add a Site or Rule, and select Rule.
 - For inbound QoS, choose Configure > Networking > Inbound QoS, and select Add a Rule.
2. Specify SSL in the Application field.
The Common Name field opens.
 3. In the Common Name field, specify the common name of a certificate.

Figure 5-43. SSL Common Name Matching Configuration



Name:

Insert Rule At:

Description:

For Traffic with the Following Characteristics:

Remote Subnet or [Host Label](#): Port or [Port Label](#):

Local Subnet or [Host Label](#): Port or [Port Label](#):

Protocol:

Traffic Type:

DSCP:

VLAN Tag ID:

Application:

Common Name: Use * to indicate wildcards.

You can use wildcards in the name: for example, *.mycompany.com. If you have three origin servers using different certificates (such as webmail.mycompany.com, internal.my company.com, and marketingweb.mycompany.com) on the server-side Steelhead appliances, all three server configurations can use the same certificate name *.mycompany.com.

For more details on SSL, see *Steelhead Appliance Deployment Guide - Protocols*.

Configuring QoS for PCoIP

PCoIP is a proprietary protocol for thin clients to access remote workstations and servers. PCoIP uses UDP as the transport mechanism, and each packet encodes priority information depending on the application: for example, screen refresh versus print traffic. The concept is similar to Citrix ICA per-packet prioritization.

PCoIP is available in advanced QoS mode. Unlike Citrix ICA, it does not require a packet-order queue. A PCoIP packet header has a 2-bit session priority and a 3-bit packet priority field. Steelhead appliances classify the packets based on only the packet priorities. The following table shows suggested PCoIP packet priority channel mapping.

PCoIP Packet Priority	IEEE802.1D User Priority	Traffic Type	Default PCoIP Channels
7	7	Network control	Unused
6	6	Voice (< 10 ms latency)	Audio (< 128 Kbps)
5	5	Video (< 100 ms latency)	Audio (greater than or equal to 128 Kbps)
4	4	Controlled load	Imaging, UFCC, UVChan

PCoIP Packet Priority	IEEE802.1D User Priority	Traffic Type	Default PCoIP Channels
3	3	Excellent effort	Bulk (USB, RVChan, and so on)
0	0	Best effort	Unused
2	2	Spare	Unused
1	1	Background	Unused

To enable PCoIP per-packet classification

1. Add a rule and enter PCoIP in the application field.

2. Select the PCoIP packet priorities from the drop-down lists.

Figure 5-44. PCoIP Configuration

QoS Sites and Rules:

▼ Add a Site or Rule — Remove Sites or Rules ↑↑ Move...

Add a: ☐ Site ☒ Rule

Name:

Description:

Parent Site:

Insert Rule At:

For Traffic with the Following Characteristics:

Local Subnet or **Host Label**: Port or **Port Label**:

Remote Subnet or **Host Label**: Port or **Port Label**:

Protocol:

VLAN Tag ID:

DSCP:

Traffic Type:

Application:

Apply these QoS Settings:

Service Class: (for non-prioritized traffic)

DSCP:

	Service Class	DSCP
PCoIP Packet Priority 6-7:	<input type="text" value="Default-Site\$\$Best-Effort"/>	<input type="text" value="Inherit from Service Class"/>
PCoIP Packet Priority 5:	<input type="text" value="Default-Site\$\$Best-Effort"/>	<input type="text" value="Inherit from Service Class"/>
PCoIP Packet Priority 4:	<input type="text" value="Default-Site\$\$Best-Effort"/>	<input type="text" value="Inherit from Service Class"/>
PCoIP Packet Priority 0-3:	<input type="text" value="Default-Site\$\$Best-Effort"/>	<input type="text" value="Inherit from Service Class"/>

Apply these Path Selections:

Path preference order (only one path will be used):

Path 1: DSCP:

Path 2: DSCP:

Path 3: DSCP:

If no configured path is up:

☒ Relay traffic from the interface normally.

☐ Drop traffic.

Configuring QoS for SnapMirror

This section describes how to configure QoS for the NetApp SnapMirror utility. In a SnapMirror environment, some data replication jobs can be more important than others. QoS for SnapMirror enables the Steelhead appliance to enforce relative priorities between replication jobs. You can distinguish replication job priorities by filer or volume.

For more details on SnapMirror, see [“SnapMirror Optimization” on page 353](#) and the *Steelhead Appliance Management Console User’s Guide*.

In certain SnapMirror environments, the data replication traffic shares a WAN link with other enterprise protocols. You can shape or rate-limit the data replication traffic with regard to other traffic. SnapMirror by itself enables you to configure different bandwidth limits for each replication job, but these are static limits that might not optimally use the available bandwidth. Riverbed QoS for SnapMirror provides a better solution for these use cases. Riverbed QoS enables QoS classification of optimized SnapMirror flows.

To configure QoS for SnapMirror

1. From the Steelhead Management Console, choose Configure > Optimization > SnapMirror.
2. Check Enable SnapMirror.
3. Add a filer and specify the filer default SnapMirror priority.

Figure 5-45. Adding a Filer

Filers and Volumes:

▼ Add a New Filer or Volume/Qtree — Remove Selected Filers and Volumes/Qtrees

Add a Filer Add a Volume/QTree

Filer Name: SnapMirrorFiler1

IP Addresses:

Filer Default Optimization Policy: SDR-Default

Filer Default SnapMirror Priority: High

Description:

Add

Name	IP Addresses	Optimization Policy	SnapMirror P
No filers or volumes/qtrees.			

The default priority for all SnapMirror volumes and qtrees is *high*. There are five priority levels to choose from. You can also choose *No Setting*, which means you have not assigned a priority and the QoS default rule applies.

4. Add a Volume/QTree and specify the SnapMirror priority.

Figure 5-46. Adding a Volume/QTree

Filers and Volumes:

▼ Add a New Filer or Volume/Qtree — Remove Selected Filers and Volumes/Qtrees

Add a Filer Add a Volume/QTree

Volume Name:

Filer:

Optimization Policy:

SnapMirror Priority:

Description:

Add

	Name	IP Addresses	Optimization Policy	SnapMirror Priority
<input type="checkbox"/>	SnapMirrorFiler1	10.10.10.1	SDR-Default	High

5. After you have completed [Step 4](#) through [Step 4](#), from the choose Configure > Network > Outbound QoS (Basic or Advanced) to configure QoS.

Basic QoS allows for automatic classification of all SnapMirror traffic into a specified class or for DSCP marking. You can use Advanced QoS to specify a class or DSCP per SnapMirror priority.

6. Enable QoS Marking.
7. To add SnapMirror:

- For Basic QoS, select the Applications tab and select Add Application. Add SnapMirror as the application and specify the class or the DSCP setting.

Figure 5-47. Adding SnapMirror to Basic QoS

The screenshot shows the 'Applications' tab in the Steelhead Configuration Manager. The 'Add Application' dialog is open, showing the configuration for 'SnapMirror'. The 'Name' field is 'SnapMirror'. The 'Position' is set to 'Start'. Under 'For Traffic with the Following Characteristics:', the 'Local Subnet or Host Label' and 'Remote Subnet or Host Label' are both '0.0.0.0/0', and 'Port or Port Label' is 'all'. The 'Protocol' is 'All', 'VLAN Tag ID' is 'all', and 'DSCP' is 'All'. The 'Traffic Type' is 'Optimized' and the 'Application' is 'SnapMirror'. Under 'Apply these QoS Settings:', the 'Service Class' is 'Normal' and 'DSCP' is 'Inherit from Service Class'. Under 'Apply these Path Selections:', three paths are listed, all with 'DSCP: Inherit DSCP from Rule'. The 'Add' button is at the bottom.

- For Advanced QoS, select QoS Sites and Rules, select Rule, and add a SnapMirror rule to the appropriate site. From the drop-down list, select the class or DSCP setting per SnapMirror priority.

Figure 5-48. Adding SnapMirror to Advanced QoS

QoS Sites and Rules:

▼ Add a Site or Rule — Remove Sites or Rules ⇅ Move...

Add as: ☐ Site ☒ Rule

Name: SnapMirror

Description:

Parent Site: Default-Site

Insert Rule At: Start

For Traffic with the Following Characteristics:

Local Subnet or Host Label: 0.0.0.0/0 Port or Port Label: all

Remote Subnet or Host Label: 0.0.0.0/0 Port or Port Label: all

Protocol: All

VLAN Tag ID: all

DSCP: All

Traffic Type: Optimized

Application: SnapMirror

Apply these QoS Settings:

Service Class: Default-Site\$\$Best-Effort

DSCP: Inherit from Service Class

	Service Class	DSCP
Highest SnapMirror Priority:	Default-Site\$\$Business-Critical	Inherit from Service Class
High SnapMirror Priority:	Default-Site\$\$Normal	Inherit from Service Class
Medium SnapMirror Priority:	Default-Site\$\$Normal	Inherit from Service Class
Low SnapMirror Priority:	Default-Site\$\$Low-Priority	Inherit from Service Class
Lowest SnapMirror Priority:	Default-Site\$\$Best-Effort	Inherit from Service Class

Apply these Path Selections:

Path preference order (only one path will be used):

Path 1: -- DSCP: Inherit DSCP from Rule

Path 2: -- DSCP: Inherit DSCP from Rule

Path 3: -- DSCP: Inherit DSCP from Rule

If no configured path is up:

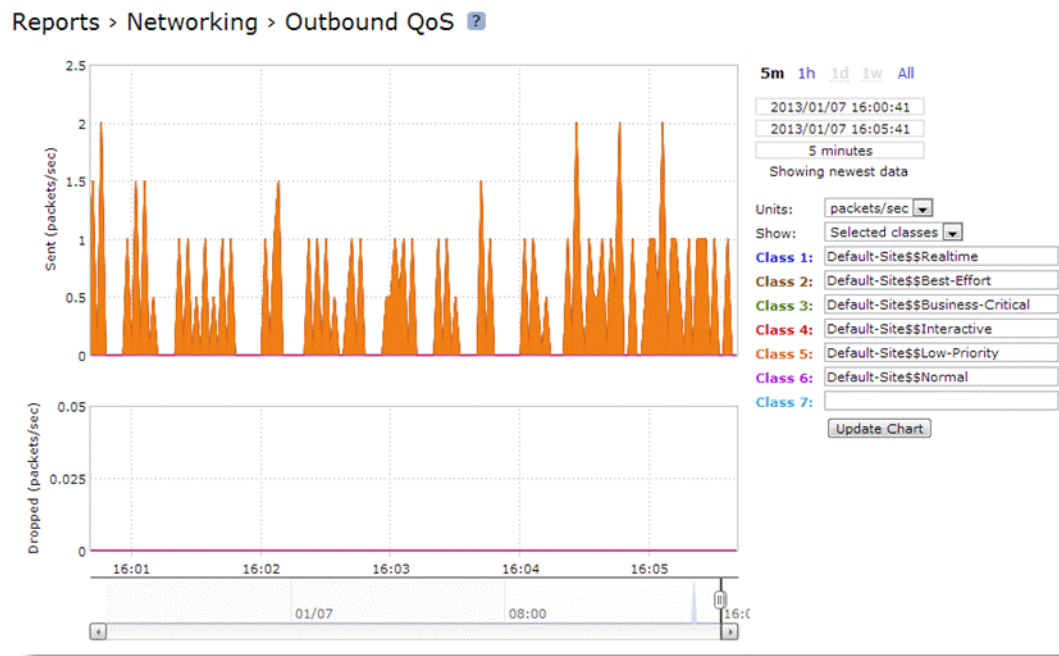
☒ Relay traffic from the interface normally.

☐ Drop traffic.

Add

8. To verify that the SnapMirror replication traffic is exiting the Steelhead appliance in the QoS class you configured, choose Reports > Networking > Outbound QoS.

Figure 5-49. SnapMirror validation



Use the following guidelines when you configure QoS for SnapMirror:

- SnapMirror QoS is supported for only outbound QoS.
- SnapMirror QoS relies on SnapMirror optimization, and therefore is supported only for optimized flows.
- QoS classification based on SnapMirror priority is supported in only advanced QoS.
- RiOS v8.5 supports NetApp Data ONTAP v7 or Data ONTAP v8 operating in 7-Mode.

CHAPTER 6 Path Selection

This chapter describes path selection. Path selection is available in RiOS v8.5 or later. Path selection enables the Steelhead appliance to redirect traffic to a predefined available WAN path for a given application in real-time, based on availability. This chapter includes the following sections.

- [“Overview of Path Selection” on page 151](#)
- [“Path Selection Implementation” on page 152](#)
- [“Site Default Path” on page 156](#)
- [“Configuring Riverbed Path Selection” on page 157](#)
- [“Valid Path Selection Deployment Design Examples” on page 159](#)
- [“Path Selection and Virtual In-Path Deployment” on page 166](#)
- [“Design Validation” on page 167](#)
- [“Design Considerations” on page 169](#)

Due to the effect of the path selection feature on packet path manipulation, Riverbed recommends that you contact your Riverbed account team to qualify all path selection design.

For more information about path selection, including configuration, see the *Steelhead Appliance Management Console User’s Guide* and the *Riverbed Command-Line Interface Reference Manual*.

Overview of Path Selection

Path selection is a RiOS technology commonly known as *intelligent dynamic WAN selection*. You can use path selection to define a specific WAN gateway for certain traffic flows, overriding the original destined WAN gateway.

WAN egress control is a transparent operation to the client, server, and any networking devices such as routers or switches. When you configure path selection, the Steelhead appliance changes the outbound physical interface transparently to the client, the server, and the intended default routing. This granular path manipulation enables you to better use and more accurately control traffic flow across multiple WAN circuits.

You must know the following nomenclature prior to reading the information in this chapter:

- **Default path** - The client default gateway (the original path), as intended by the end client or routed LAN.

- **Primary path selection path** - The first and preferred path that you configure using path selection. The primary path can also be the same path as the default gateway.
- **Secondary and tertiary path selection path** - In case the configured primary path is unavailable, you can configure two alternate paths.
- **Site default path** - A list of default paths that you configure by path priority, for packets destined to a particular site. You can use these paths prior to completion of classification or for any flows not matching an explicit rule. For more details on site default path, see [“Site Default Path” on page 156](#).

Path Selection Implementation

This section includes the following topics:

- [“Path Selection Properties” on page 152](#)
- [“Example Path Selection Implementation” on page 153](#)
- [“Identifying Traffic Flow Candidates” on page 154](#)

Path Selection Properties

Path selection is a combination of the following operations:

- **Path availability** - Ability to identify and logically configure a WAN path. Path availability includes additional parameters to ensure the viability of the path.
- **Application steering** - An operation that combines the QoS packet-flow classification with path control. Path selection can steer different classes of traffic through the different primary, secondary, tertiary, or site default paths that you have configured in path availability.

The object of path availability is to construct a logical medium that you want to use for traffic flow. The construction of the path is not adding new IP addresses or making physical connections. Path availability is defining an egress point and manipulating the direction the egressing packets must go.

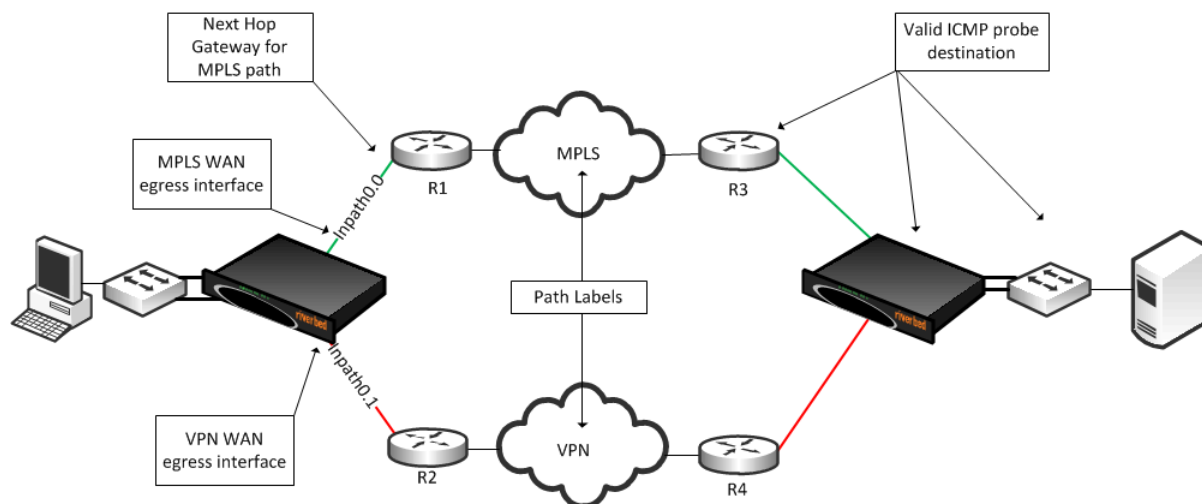
The following properties apply to a path you configure using path selection:

- Path label
- Next-hop gateway
- ICMP probe target IP address
- Tunnel mode
- Probe time out
- Probe threshold
- Probe DSCP
- Egress interface

Example Path Selection Implementation

Figure 6-1 shows an example using the properties of path selection. The example shows a dual-homed site with a MPLS WAN link provided by a carrier, labeled MPLS. The second link is a public Internet connection, labeled VPN.

Figure 6-1. Path Selection Properties Example



Each path is probed for availability based on a schedule with default configuration of two seconds. The probe transmits an ICMP request from the configured in-path interface toward the probe destination IP, and after receipt of an ICMP response, the path is declared available for use. A path is determined down after the count of consecutive probe failures surpasses the configured probe threshold. The default threshold is three probe packets.

Note that the Steelhead appliance is looking for an ICMP response from the probe destination to determine path availability. Even if the ICMP response traverses unintended devices or WANs, the path is available as long as the configured in-path interface receives the ICMP response. This can result in false positive path availability. The example shown in Figure 6-1 assumes that the MPLS path is configured with `inpath0_0`, along with a probe destination of Router 1. Even if the MPLS network fails, the path remains up as long as Router 1 continues to send ICMP responses to the Steelhead appliance `inpath0_0`. Likewise, assume that the MPLS path, `inpath0_0`, is configured to send probes to Router 3 as the probe destination.

The MPLS fails, Router 1 forwards the ICMP request to Router 2, across the VPN, through Router 4, and on to Router 3. Router 3 can respond, sending ICMP responses down and back over the VPN, reaching the Steelhead appliance `inpath0_0`. In either case, the MPLS path availability remains connected, though the likely intention is that the path shows it is not connected when the MPLS WAN is down.

Riverbed recommends that you locate an address on the remote side of the path, and make sure devices in the path treat the probe as expected during a failure. If the MPLS path has `inpath0_0` configured with a probe destination of Router 3 and a next-hop gateway of Router 1, then configure Router 1 in which traffic to Router 3 can only go over the MPLS network. If the MPLS network fails, then configure Router 1, or another device, to drop the ICMP request probe from `inpath0_0` to Router 3. An appropriate probe destination for a path can be a remote router loopback address, or one of the remote Steelhead in-path interfaces.

In RiOS v8.6 and later, you can configure the Steelhead appliance on a per-path basis to perform firewall traversal using GRE encapsulation to a remote Steelhead appliance peer. The Steelhead appliances use the configured destination IP for the probe as the other endpoint when you set the tunnel mode setting in a path to GRE.

Each path has its own independent IP address to probe, yet this address can be the identical one for each path. Therefore, each path can poll on the same probe destination. Note that the ICMP request has to use whichever physical interface selected for the path. In [Figure 6-1](#), the MPLS path egresses its packets using the inpath0_0 interface, hence all traffic uses the corresponding WAN0_0 interface. Meanwhile, the VPN path egresses its packets using in-path0_1, hence all traffic uses the corresponding WAN0_1 interface.

The next-hop gateway serves the following purposes for the path selection design because the gateway provides the new routing path for packets to travel through on the way to their destination:

- Replaces the destination MAC address of packets with the MAC address of the alternate gateway. The gateway MAC address is learned by the Steelhead appliance in-path interface. As part of steering packets, the destination MAC address of the packets is altered to match learned MAC address of the configured new gateway.

Path selection requires a Layer-2 connection with the gateway; the path cannot be across a routed link. This is referred to as *Layer-2 redirect by next hop MAC*.

- Switches the outbound interface from the original in-path interface with the desired primary path.

The path selection solution is implemented completely transparently, regardless of existing routing metrics. The primary path selection path gateway accepts the steered packets and proceeds to forward them onto the corresponding WAN.

- The Steelhead appliance takes no action in having to reconfigure the Layer-3 routing parameters of the routers in the network.

The Steelhead appliance never takes an action to inject any routes or alter the routing instances. The traffic source whose packets are sent to the primary path selection gateway have no visibility into the changes the Steelhead appliance applies. Therefore, the client (or server) continues to send any and all packets to the gateway address they are configured with. This is referred to as *Layer-2 redirection by interface*.

You can configure a maximum of 1024 paths per Steelhead appliance. You can select the same in-path interface for multiple path configurations if your network requires this implementation.

WAN interface selection is based on identified traffic type and availability of the end-to-end path, depending on how you configure your Steelhead appliance. Note that path selection remains functional even if you pause the optimization service, or if the optimization service becomes unavailable. If the Steelhead appliance fails completely, then path selection is no longer applicable and traffic proceeds as normal, following its default gateway.

Identifying Traffic Flow Candidates

The critical step for path selection is to identify traffic flow and to associate these traffic flow candidates with a different, configured path. In this step, the QoS functionality interacts with the path selection feature. Use the following methods to identify traffic that can benefit from path selection:

- The AFE can help you to identify the traffic and steer the traffic along a configured path. For more information about AFE, see [“Application Flow Engine” on page 83](#).

Some limitations exist when you use AFE in conjunction with path selection. AFE, or any deep packet inspection technique, requires examining several of the beginning packets of a connection before it can identify the traffic. That means that after the beginning packets have been identified, they can be sent on a path different than the one you chose. This midstream switching has implications in various environments involving firewalls and dual Internet egress environments. For more details, see [“Firewall Path Traversal Deployment” on page 164](#) and [“Design Considerations” on page 169](#).

- You can also use IP header information as an alternate method for identifying traffic. IP header information identification consists of any of the following combinations:
 - Source IP
 - Destination IP
 - Source port address
 - Destination port address
 - DSCP mark
 - VLAN tag
 - Optimized/Unoptimized traffic
 - Layer-4 protocols (TCP, UDP, GRE, and so on)

For each application rule you configure, you can add a maximum of three different paths: primary path, secondary path, and tertiary path. The paths you choose cascade from one to the next, based on availability.

You have the option to drop the traffic if no alternate path is available. Dropping traffic is useful when you prefer not to use bandwidth on the secondary (or tertiary) path selection path circuit in case of failure on the primary selected path. If you choose not to override the original intended route, then traffic is relayed normally. The traffic continues to flow normally along the original intended path, following the default gateway.

Traffic identification and path steering is independent of optimized versus pass-through traffic. Path selection takes action on the configured traffic, no matter the optimization status of the traffic.

Path selection configuration is also independent of any QoS settings. This means that you can apply path selection rules with and without having to enable QoS marking and/or shaping. Path selection uses QoS features, and you must complete a subset in the QoS configuration page when you configure path selection. Path selection uses the same rule sets as QoS, therefore it does not increase the rule count number against the model limit as specified for QoS.

For example:

- **With path selection enabled and QoS disabled-** You configure ten applications to identify and steer. The total is ten.
- **With path selection enabled and QoS enabled -** Assume you build 100 classes and 100 rules, equally 200 total. If 50 of those rules have path selection steering paths, the total still remains 200.

In other words, you can look at path selection as an extra policy similar to a priority setting within a rule.

Remember that return traffic in path selection is not influenced or manipulated in any way to take the steered path from the sending Steelhead appliance. You must install and configure a remote Steelhead appliance with the appropriate path selection configuration, and steer the return traffic on the same path.

Site Default Path

The site default path feature is an optional configuration you can use when path selection is enabled. The site default path feature is a per-site configuration (including the 0.0.0.0/0 default site) in the QoS configuration pages of the Steelhead Management Console.

Note: When you use AFE with path selection, the beginning packets of the traffic flow take the default path or rule header that matches the traffic.

The site default path provides a default path for packets destined to a particular site for traffic that has not received an explicit, specific rule. The site default path can be identical to the configured path selection (steered path) or another path.

An example use case of the site default path feature is when you want to steer the OOB splice traffic along with any traffic not configured with a specific rule (Figure 6-2).

Figure 6-2. Site Default Path Example

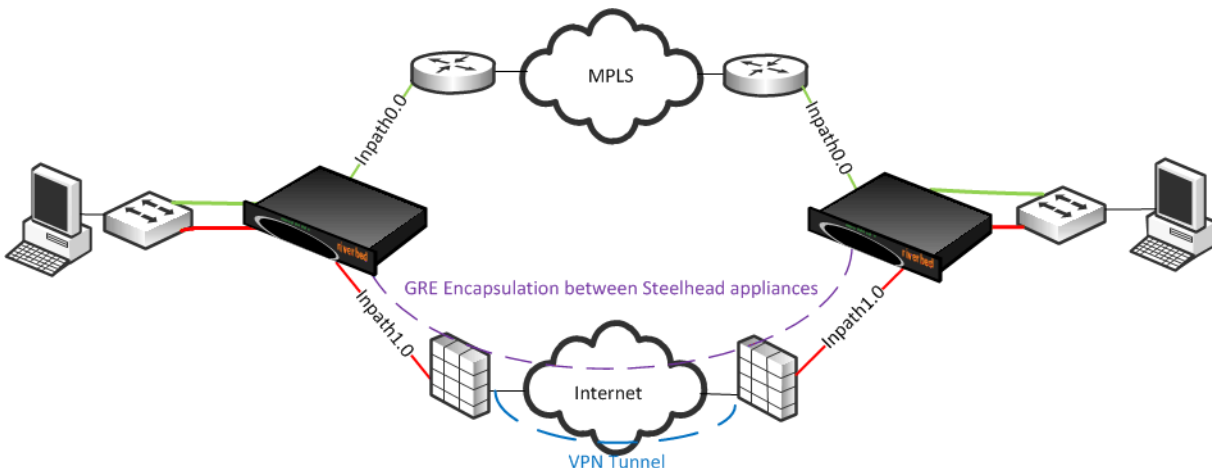


Figure 6-2 shows a typical branch site in which a Steelhead appliance is installed on the LAN side of a router and firewall. Each WAN-edge device connects to a separate cloud; the router is connected to a private MPLS provider, and the firewall has a direct connection to the open public Internet. For the purposes of the example, assume that the firewall has been configured with a VPN tunnel back to the data center to maintain dual separate paths of connectivity for the VPN and MPLS.

Standard path selection configuration applies throughout the entire process:

- Configure each path, including their corresponding physical interface and gateways: VPN and MPLS.
- Configure the ICMP probe destination IP address and enable path selection.
- For each QoS configured site (including the default “0.0.0.0/0”), select a site default path. This is the firewall path. Complete this step in the QoS configuration pages of the Steelhead Management Console.
- In the QoS configuration screen, select specific applications with their preferred primary and secondary path.

Path steering does not take place until you identify and configure the application as a rule. Because identifying every application is not optimal, you can use the site default path rule for any traffic that does not have a specific rule: for example, the OOB splice traffic.

Note: Steelhead appliances support paths with stateful security appliances if you use header-only rules, or if you use the firewall path traversal capability. For details see [“Firewall Path Traversal Deployment” on page 164](#).

Configuring Riverbed Path Selection

This section describes the basic steps for configuring path selection using the Steelhead Management Console. This section also includes a configuration example. For more details on the Management Console, see the *Steelhead Appliance Management Console User's Guide*.

You can also use the Riverbed CLI to configure path selection. For more information about path selection commands, see the *Riverbed Command-Line Interface Reference Manual*.

You can use the CMC to enable path selection and to configure and apply path selection rules to multiple Steelhead appliances. For details, see the *Riverbed Central Management Console User's Guide*.

To perform the basic steps to configure path selection

1. Install RiOS v8.5 or later.
2. Build the various paths available. Choose Configure > Networking > Path Selection, and select Add a New Path.

Figure 6-3. Add a New Path Page

Configure > Networking > Path Selection ?

Path Selection Settings

☒ Enable Path Selection

Apply

Path Settings:

▼ Add a New Path — Remove Selected Paths

Path Definition:

Name:

Gateway IP Address:

Interface:

Tunnel Mode:

Probe Packet Settings:

Remote IP Address:

DSCP:

Timeout: seconds

Threshold: packets

Add

3. Specify a name for the path
4. Specify the gateway IP address for this specific path.
5. From the Interface drop-down list, select the interface.
6. Specify a remote IP address to poll using ICMP probes:
 - You can select a specific DSCP mark for the ICMP probes. You must select this option if the service providers are applying QoS metrics based on DSCP marking, and each provider is using a different type of metrics.
 - You can alter the timeout and threshold. The default is set to two seconds for timeout, with a threshold of three successive failed packets.
7. Select GRE from the Tunnel Mode drop-down list if you want encapsulation to the remote IP address.
The Steelhead appliance at the remote site must be running RiOS v8.6 or later, with path selection configured.
8. Select Enable Path Selection.
9. Click **Apply**.
You do not need to restart the Steelhead appliance to enable path selection. At this point, path selection is enabled and you have configured the different available paths.
10. Associate an application rule with the paths in which you want to steer the traffic.

Complete this step in the QoS configuration pages (either basic or advanced) of the Steelhead Management Console. You must follow the general application classification configuration as it pertains to prioritization—except for the added steps to select the primary, secondary, and tertiary path—to steer that traffic onto as identified by the AFE and/or IP header-based (Layer-3/4) characteristics.

Figure 6-4. Configuring Paths in the Outbound QoS Add Application Page

The screenshot shows the 'Add Application' page in the Steelhead Management Console. The page has three tabs: 'Sites', 'Applications' (selected), and 'Service Policies'. Below the tabs are three buttons: 'Add Application' (with a dropdown arrow), 'Remove Application', and 'Move Application' (with up/down arrows). The main form area is divided into several sections:

- Name:** A text input field with a red dot icon.
- Description:** A text input field.
- Position:** A dropdown menu with 'End' selected.
- For Traffic with the Following Characteristics:**
 - Local Subnet or Host Label:** Text input '0.0.0.0/0'.
 - Remote Subnet or Host Label:** Text input '0.0.0.0/0'.
 - Port or Port Label:** Two dropdown menus, both with 'all' selected.
 - Protocol:** Dropdown menu with 'All' selected.
 - VLAN Tag ID:** Text input 'all'.
 - DSCP:** Dropdown menu with 'All' selected.
 - Traffic Type:** Dropdown menu with 'All' selected.
 - Application:** Text input field.
- Apply these QoS Settings:**
 - Service Class:** Dropdown menu with 'Realtime' selected.
 - DSCP:** Dropdown menu with 'Inherit from Service Class' selected.
- Apply these Path Selections:**
 - Path preference order (only one path will be used):**
 - Path 1:** Dropdown menu with '--' selected. **DSCP:** Dropdown menu with 'Inherit DSCP from Application' selected.
 - Path 2:** Dropdown menu with '--' selected. **DSCP:** Dropdown menu with 'Inherit DSCP from Application' selected.
 - Path 3:** Dropdown menu with '--' selected. **DSCP:** Dropdown menu with 'Inherit DSCP from Application' selected.
 - If paths are configured and all down:**
 - ☒ Relay traffic from the interface normally.
 - ☐ Drop traffic.

At the bottom left of the form is an 'Add' button.

Valid Path Selection Deployment Design Examples

The section shows valid path selection deployment examples. The examples in this section show only one side of the WAN. You must assume that the remote side also has similar path selection capabilities and configurations for symmetric for return traffic. This section includes the following topics:

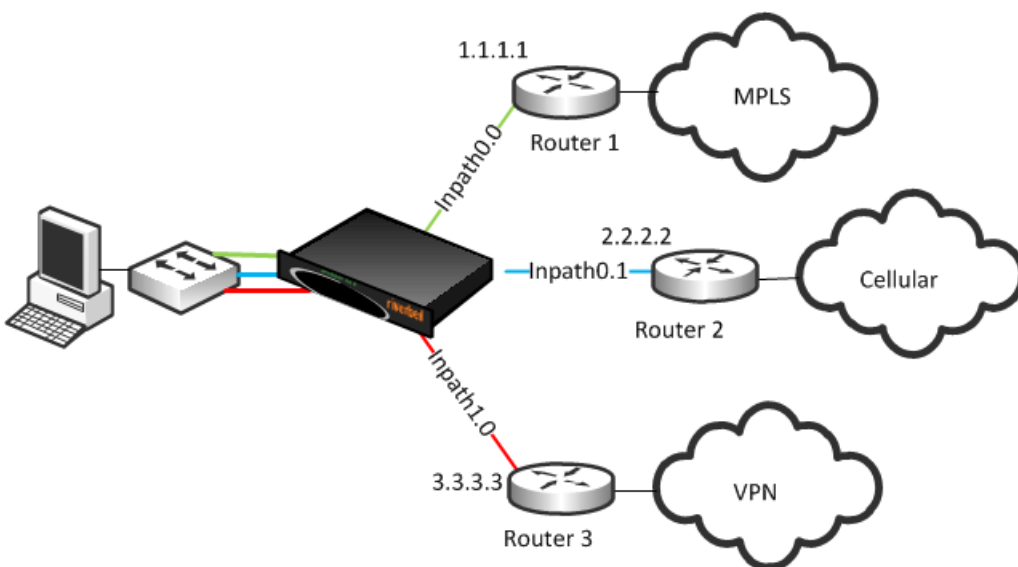
- [“Basic Multiple Route Path Deployment” on page 160](#)

- [“Complex Parallel Path Deployment” on page 162](#)
- [“Complex Single In-Path Interface Deployment” on page 163](#)
- [“Serial Deployment” on page 164](#)
- [“Firewall Path Traversal Deployment” on page 164](#)

Basic Multiple Route Path Deployment

Figure 6-5 shows a Steelhead appliance connected to three separate routers on three distinct in-path connections. Inpath0_0 is connected to Router 1, which is serving an MPLS connection. Inpath0_1 is connected to Router 2, which is serving a cellular-based WAN connection. Inpath1_0 is connected to Router 3, which is serving a VPN-based connection.

Figure 6-5. Basic Multiple Route Path Deployment



To configure path selection on the Steelhead appliance as shown in [Figure 6-5](#)

1. From the Management Console, choose Configure > Networking > Path Selection.
2. Create a separate path for each of the three different carriers, labeling each path with the proper router gateway IP address and remote (probe) IP address, as shown next.

For details on configuring path selection, see [“Configuring Riverbed Path Selection” on page 157](#).

Figure 6-6. Three Path Selection Paths

Path Settings:

[+ Add a New Path](#) [- Remove Selected Paths](#)

<input type="checkbox"/>	Name ⇅	Remote IP ⇅	Gateway IP ⇅	Interface ⇅	Status ⇅	Probe DSCP ⇅	Tunnel Mode ⇅
<input type="checkbox"/>	Cellular_3G	172.30.1.1	2.2.2.2	inpath1_0	Down	0	None
<input type="checkbox"/>	mpls	8.8.8.8	192.168.1.249	inpath0_0	Up	0	None
<input type="checkbox"/>	vpn	8.8.8.8	3.3.3.3	inpath1_0	Down	0	None

3. Select Enable Path Selection.

4. Choose Configure > Networking > Outbound QoS (Advanced).
5. Configure the application with the desired path order.

Figure 6-7 shows an application of type *Exchange* is selected to be steered through the VPN path, followed by MPLS if VPN is not available, and then finally over the cellular link.

Figure 6-7. Configuring Path Selection for Application Exchange

20 **Exchange** Normal

Name: Exchange

Description:

For Traffic with the Following Characteristics:

Local Subnet or **Host Label**: 0.0.0.0/0 Port or **Port Label**: all

Remote Subnet or **Host Label**: 0.0.0.0/0 Port or **Port Label**: all

Protocol: All

VLAN Tag ID: all

DSCP: All

Traffic Type: All

Application: Exchange

Apply these QoS Settings:

Service Class: Normal

DSCP: Inherit from Service Class

Apply these Path Selections:

Path preference order (only one path will be used):

Path 1: vpn DSCP: Inherit DSCP from Application

Path 2: mpls DSCP: Inherit DSCP from Application

Path 3: Cellular_3G DSCP: Inherit DSCP from Application

If paths are configured and all down:

☒ Relay traffic from the interface normally.

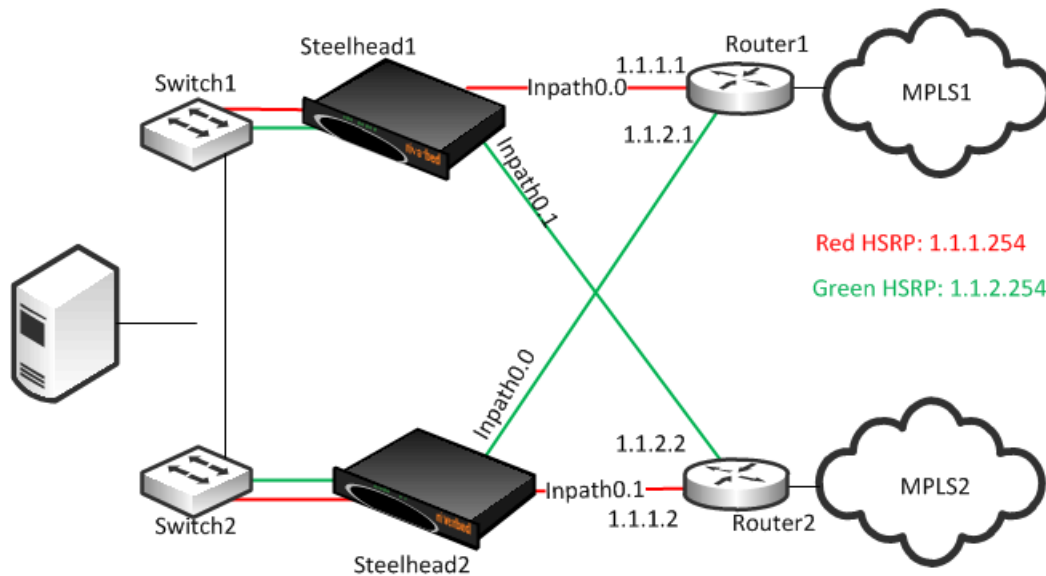
☐ Drop traffic.

Apply

Complex Parallel Path Deployment

Figure 6-8 shows a dual parallel Steelhead appliance deployment on the WAN side. On the WAN side, Router 1 connects to the MPLS1 provider, and Router 2 connects to the MPLS2 provider. On the LAN side, each switch has a connection to both routers.

Figure 6-8. Complex Parallel Path Deployment



While each of the links in Figure 6-8 can also be individual Layer-3 links, in this example there are two networks with HSRP configured on each network. When you define paths, use the real IP addresses of the routers as the gateway, and not the virtual IP address. If you use a virtual IP address, you can cause the gateway to reside on the LAN side of the Steelhead appliance in-path interface, resulting in unintended traffic flow. If your design is configured with a single HSRP group covering both routers, you must use the real IP address of the router as the gateway, and not the virtual IP address.

Each Steelhead appliance is connected to each router with the MPLS provider; therefore, both Steelhead appliances can make uniform path selection decisions, with traffic moving toward the same router and provider.

Riverbed recommends that you configure both Steelhead appliances with equivalent paths, uniform QoS classification, and the same path selection logic. This example shows Steelhead appliance configured with two paths: MPLS1 and MPLS2.

Figure 6-9 shows the path configuration of Steelhead1 from Figure 6-8.

Figure 6-9. Steelhead1 Path Configuration

Path Settings:							
+ Add a New Path		- Remove Selected Paths					
<input type="checkbox"/>	Name ⇅	Remote IP ⇅	Gateway IP ⇅	Interface ⇅	Status ⇅	Probe DSCP ⇅	Tunnel Mode ⇅
<input type="checkbox"/>	mpls1	8.8.8.8	1.1.1.1	inpath0_0	Down	0	None
<input type="checkbox"/>	mpls2	8.8.8.8	1.1.2.2	inpath1_0	Down	0	None

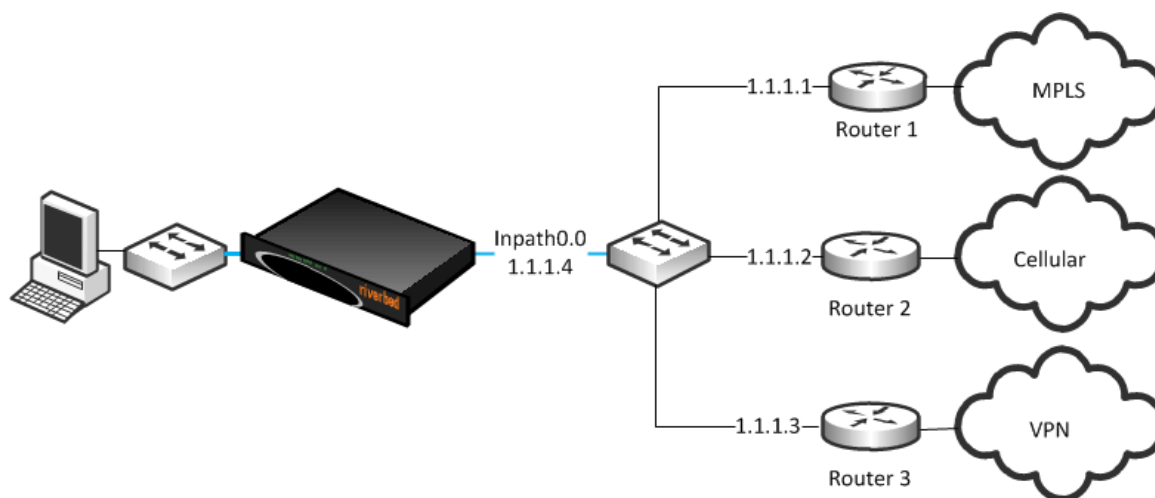
Steelhead2 is configured with the equivalent paths, but with the respective gateway of Router1 IP address 1.1.2.1 and Router2 IP address 1.1.1.2.

If your design is comprised of a single HSRP group covering both routers, configure Steelhead2 so that it is identical to that shown in [Figure 6-9](#), in which the path gateway references the real IP interface and not the virtual IP.

Complex Single In-Path Interface Deployment

[Figure 6-10](#) shows the Steelhead appliance connected through a single in-path interface connection, but the WAN side is composed of multiple WAN routers, each to their own separate provider. Note that the LAN side of the routers all share the same IP segment. Because they share the same IP segment, achieving path selection configuration in this setup is valid, because the Steelhead appliance can be configured with different gateway addresses traversing the same in-path interface.

Figure 6-10. Complex Single In-Path Interface Deployment



[Figure 6-11](#) represents the path configuration for the deployment shown in [Figure 6-10](#). Each path has the same inpath0_0 as the egress interface, but the gateway IP differs.

Figure 6-11. Path Configuration for Single In-Path Interface Deployment

Path Settings:

[+ Add a New Path](#) [- Remove Selected Paths](#)

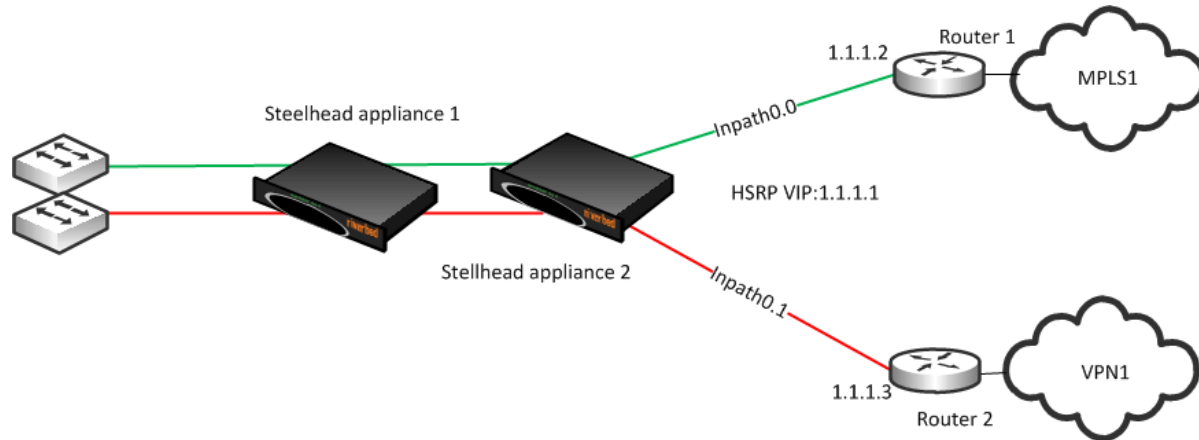
<input type="checkbox"/>	Name ↕	Remote IP ↕	Gateway IP ↕	Interface ↕	Status ↕	Probe DSCP ↕	Tunnel Mode ↕
<input type="checkbox"/>	Cellular_3G	8.8.8.8	1.1.1.2	inpath1_0	Down	0	None
<input type="checkbox"/>	mpls	8.8.8.8	1.1.1.1	inpath0_0	Down	0	None
<input type="checkbox"/>	vpn	8.8.8.8	1.1.1.3	inpath1_0	Down	0	None

Important: A complex single in-path interface deployment is valid for path selection when all the routers are on the same subnet as the Steelhead appliance single in-path interface. If the in-path interface is on an 802.1Q trunk, you cannot use path selection to direct traffic to different routers on different VLANs. The switch shown in [Figure 6-10](#) is a Layer-2 switch, and therefore path selection can make the decision to send traffic to the appropriate router MAC address.

Serial Deployment

Figure 6-12 shows a dual serial Steelhead appliance deployment. Steelhead appliance 1 is the client Steelhead appliance, and Steelhead appliance 2 is referred to as the *middle file engine* (MFE). On the WAN, Router 1 is connected to the MPLS provider, and Router 2 connects to the customer internal network using a VPN connection. In this example, Riverbed recommends that you use the real IP address of the router as the path gateway instead of the virtual IP provided by HSRP and VRRP.

Figure 6-12. Serial Deployment



You can use path selection in a serial deployment when:

- both Steelhead appliances have identical path selection configuration.
- if correct addressing is in use, you must configure Steelhead appliance 2 to relay the inner channel of Steelhead appliance 1.
- if you are using Full Transparency, you must use the **path-selection settings bypass non-local-trpy enable** command on Steelhead appliance 2.

Firewall Path Traversal Deployment

This section describes how to configure firewall path traversal deployment. This section contains the following topics:

- [“MTU and MSS Adjustment When Using Firewall Path Traversal” on page 165](#)
- [“Firewall Path Traversal Deployment Example” on page 166](#)

Stateful firewall devices typically provide security services including:

- tracking the TCP connection state.
- blocking a sequence of packets.

Stateful security devices add a level of complexity to path selection environments when the Steelhead appliance attempts to make any path changes to a connection midstream. The most common examples of midstream switching are:

- failure of a higher-priority path, failing to firewall path.
- recovery of path, resuming traffic to a firewall path.
- using AFE for identification because the first packets of a connection are not recognized yet and can traverse a default path.

When a path changes midstream, the stateful firewall device is likely to see only some or none of the packets necessary to keep state and sequence numbers. When receiving packets perceived to be out of order or belonging to a connection with inaccurate state information, stateful firewalls generally drop these packets.

Beginning with RiOS 8.6, Riverbed recommends that you use the firewall path traversal capability to leverage GRE encapsulation over paths traversing a stateful firewall. When you use standard GRE between Steelhead appliances, connections can be switched midstream because the firewall only detects the encapsulated packets. You can configure GRE encapsulation per path, and when enabled, the Steelhead appliance attempts to encapsulate packets to the remote Steelhead appliance at the configured remote IP. You can use multiple paths using GRE encapsulation between the same Steelhead appliances, and the original packet or QoS-configured DSCP values are reflected in the GRE packets.

Note: There is a loss of visibility on the firewall when you use GRE encapsulation. You also might need additional configuration on the firewall to allow the GRE packets between Steelhead appliances.

MTU and MSS Adjustment When Using Firewall Path Traversal

When you use GRE encapsulation, consider that there is an additional 24-byte overhead added to packets. This overhead can cause fragmentation of large packets, because the extra added bytes cause the packet to exceed the maximum transmission unit (MTU) configured in the network. Fragmentation has the negative effect of sending inefficiently-sized packets, and dropping packets that might have the *do not fragment* option set.

You can prevent fragmentation by adjusting the maximum TCP payload, or MSS value, to account for the overhead added by GRE. When you configure a path with the tunnel mode set to GRE, the Steelhead appliance measures to reduce potential fragmentation for TCP traffic.

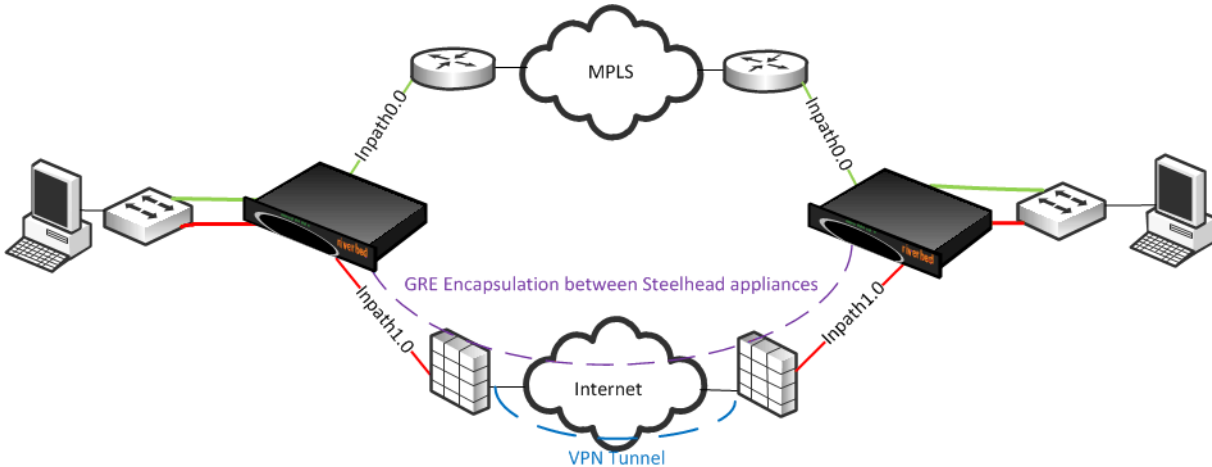
This automatically applied MSS value ensures that in most environments, TCP packets are not fragmented, even with the additional GRE overhead. In an optimized case, the client and server connections with the Steelhead appliance are not impacted by the MSS adjustment procedure. For pass-through TCP traffic, the Steelhead appliance adjust the MSS value to make room for a GRE header. To turn off this automatic MSS adjustment, use the command **no path-selection settings tunnel adjust-mss enable**.

For more information about MTU, see [“MTU Sizing” on page 445](#).

Firewall Path Traversal Deployment Example

Figure 6-13 shows a dual-parallel installation of Steelhead appliances in a dual-homed WAN scenario. In this example, a firewall is installed on the edge of the Internet path. The Steelhead appliance has visibility into both MPLS and VPN paths.

Figure 6-13. Firewall Path Deployment



There are two paths defined: the green path over MPLS and the red path traversing stateful firewall devices. The firewall path is configured for GRE encapsulation between the Steelhead appliances. When you configure a path for GRE encapsulation, it affects only the decision of the local Steelhead appliance, in that the opposing Steelhead appliance must also have similar path configuration for return traffic to be encapsulated.

Note that the Steelhead appliances are not providing the VPN functionality over the Internet, but sending traffic over the VPN tunnel provided by the existing firewalls in the path. Remember that this Steelhead appliance configuration for path traversal over firewalls uses standard GRE encapsulation, which is not a secure method of traversing the Internet.

In addition, the firewalls provide other necessary functions, such as NAT, and Riverbed does not recommend that you to use the Steelhead appliance GRE capability instead of direct VPN functionality.

Path Selection and Virtual In-Path Deployment

Riverbed recommends that you not use virtual in-path deployments for path selection, but always use physical in-path deployments. Virtual in-path deployments often have caveats that limit path selection effectiveness, including but not limited to the following:

- Typically, only traffic that is optimized is redirected to the Steelhead appliance, and therefore the Steelhead appliance is limited to identifying and acting on only that subset of total traffic. Although you can configure devices to redirect all traffic, this is often undesirable due to adding increased load and complexity.

- Additional routing devices often exist after the Steelhead appliance makes the path selection decision.

For example, consider an environment with dual Layer-3 switches and dual routers connected to different service providers. Policy-based routing (PBR) is configured on the Layer-3 switches, and the Steelhead appliances make the path selection decision about which Layer-3 switch to send traffic to. The Layer-3 switches then make an independent routing decision to send traffic to a router, and therefore provider, rendering the Steelhead appliance path selection decision meaningless.

Considering additional routing devices is important in physical in-path deployments, but holds additional weight in virtual in-path deployments because of the added restriction of only certain devices being capable of redirection. For example, many firewall devices have limited functionality when supporting virtual in-path mechanisms.

- Path selection next-hop functionality is not supported with WCCP. The Steelhead appliance cannot choose to redirect packets using a different in-path interface or to send traffic to a configured gateway. Only limited functionality is available, enabling the Steelhead appliance to mark packets with DCSP with different criteria depending on path availability.

Design Validation

In RiOS v8.5.1 and later you can use CLI commands and Steelhead Management Console report pages to verify path selection operations and to validate your path selection configuration.

For details, see the *Riverbed Command-Line Interface Reference Manual* and the *Steelhead Appliance Management Console User's Guide*.

You can use the **show** command to verify path selection settings and configuration:

```
CFE # show path-selection ?
interface      Name of the interface
path           Path name
paths          Display all configured paths
settings       Path selection settings
status         Display feature status
```

You can use the **show path-selection path ##### state** command, in which ##### is the path name, to provide path configuration and the learned MAC address of the next-hop destination to use:

```
CFE # show path-selection path mpls state
Name:          mpls
ID:            14
Status:        Up
Interface:     inpath0_0
Gateway IP     1.1.1.1
Remote IP      8.8.8.8
Probe DSCP:    0
Probe Timeout: 2 seconds
Probe Threshold 3
VLAN:          65535
Source MAC:    00:50:56:b8:1f:eb
Next Hop MAC:  00:14:6a:2b:61:c9
```

You can use the **show path-selection paths stats** command to display the ICMP probe statistics for each path:

```
Name:          WWW
ID:            15
Bytes:         0
Probe Requests 1992
Probe Requests Really Mismatch:0
Probe Ricochet Ignored: 0
```

Probe Ricochet Relay Mismatch:0

Name: mpls
 ID: 14
 Bytes: 1210619
 Probe Requests: 1992
 Probe Requests Realy Mismatch:0
 Probe Ricochet Ignored: 0
 Probe Ricochet Relay Mismatch:0

You can validate your design from the Steelhead Management Console with these reports:

- **Reports > Networking > Current Connections** - shows details per connection (Figure 6-14).
- **Reports > Networking > Path Selection** - shows you details per path (Figure 6-15).

Figure 6-14. Current Connections Report

Connection type: Failed terminated
 Connection age: 55 seconds
 Transport: TRANSPORT_ID_NONE
 Passthrough reason: No Steelhead on path to server

Path Selection
 Relayed: 251 Dropped: 0 Bypassed: 0 Reflected: 0

Path name	Status	Last started	Bytes	DSCP
(Relayed)		2013/11/25 12:02:44	251	Reflect
direct	Up	2013/11/25 12:02:44	3,900	Reflect

Gateway IP: 10.32.149.65
 Remote IP: 8.8.8.8
 Probe-DSCP: 0

Refresh Data
 Reset Connection
[Log for this Steelhead](#)

Figure 6-15. Path Selection Report

Path Selection Enabled: Yes

Paths

Filter by Status: All Up Down

Name	Status	Interface	Gateway IP	Next Hop MAC	Remote IP
Path_1	Up	inpath0_0	--	00:50:56:93:44:32	10.11.6.4
Path_2	Down	inpath0_0	--	00:00:00:00:00:00	10.12.1.18
Path_3	Up	inpath0_0	--	00:50:56:93:44:32	10.11.6.4

Probe

DSCP: 10 (AF11)
 Timeout (sec): 5
 Threshold (sec): 3
 Requests: 11
 Response Relay Mismatch: 0
 Ricochet Ignored: 0
 Ricochet Relay Mismatch: 0

Miscellaneous

Bytes: 0
 VLAN: 65535 (unknown)
 Source MAC: 00:0e:b6:91:4a:ef

Design Considerations

Consider the following when you use path selection:

- Due to the effect of the path selection feature on packet path manipulation, Riverbed recommends that you contact your Riverbed account team to qualify all path selection design.
- Path selection requires dual-ended Steelhead appliance deployments.
- You cannot use AFE for Internet-bound applications to select paths with different Internet egress points. Using AFE implies that packets prior to identification traverse one path, but that after identification, the connection can switch midstream to a different path. If these two paths use different egress points to the Internet, the packets on each path use different NAT public Internet IP addresses and appear as two different sources to the Internet server. These are examples in which multiple Internet egress can exist:

- **Direct-to-Internet at the branch office and Internet at the data center** - You cannot use AFE to decide that some Internet applications exit directly from the branch office and others from the data center.

For example, the default path that directly reaches the Internet is at the branch, but you configure AFE for Facebook traffic to traverse a path to the data center. The beginning packets of the connection exit from the branch with a externally translated address from the branch Internet provider. After identified, path selection switches midstream to the path to the data center, where the traffic is translated to a different Internet address.

- **Dual data centers, each with Internet egress** - You cannot use AFE to determine what path Internet-bound applications traverse.
- Be mindful of WAN-side routing, because it always takes precedence over path selection. Routers on the WAN side of a Steelhead appliance can always override and reroute traffic according to their configuration. Be aware of upstream router configuration, so you that avoid unintended traffic redirection. Placing the Steelhead appliance closer to the edge of a WAN helps to avoid this scenario. Some examples of this scenario include but are not limited to:
 - **WAN-Aggregation Layer** - All routers consolidate into a pair of Layer-3 switches. Path selection must occur on the WAN side of this layer. This is more common in data centers.
 - **WAN-side router with multiple circuits** - The router decides on which circuit to send traffic. You cannot use path selection effectively in this scenario.
- Path selection is not effective in any environment in which independent routing decisions are made at the site after the Steelhead appliance path selection decision has already occurred.
- Path selection in virtual in-path environments have additional considerations. For details, see [“Path Selection and Virtual In-Path Deployment” on page 166](#).
- Subnet side rules exclude subnets from changing paths.
- The Steelhead appliance does not apply path selection configuration for traffic destined onto the same IP segment as the in-path interface. This is useful for routing updates if you have deployed the Steelhead appliance in the direct path of that traffic.
- The Steelhead appliance never takes on the role of the router or of a default gateway. Because the path selection solution is transparent, you do not have to make network design changes to accommodate path selection design.
- The primary and auxiliary interfaces of a Steelhead appliance do not support path selection.

- Path selection is compatible with all virtual and physical Steelhead appliance models running RiOS v8.5 or later.
- You must disable RSP to enable path selection. Current virtualization capabilities, including VSP on Steelhead EX v2.0 and later, are compatible with path selection.
- A Steelhead appliance with path selection enabled has no enforcement on the return path.

If you want to influence the return path of traffic and override the original traffic path, you must deploy a Steelhead appliance near the return traffic WAN junction point. Traffic returning on a different path is commonly known as *asymmetric routing*. Typically, networks are not designed in this way; however if this traffic pattern exists, it might not be completely detrimental, because the Steelhead appliance can rely on existing features and complete the optimization.

For more information about asymmetric routing, see the *Steelhead Appliance Management Console User's Guide*.

- A single Steelhead appliance can maintain optimization even if traffic is received on a different in-path interface from the original sending in-path interface. Because the Steelhead appliance shares internal flow table with itself, it can complete the optimization process with no asymmetric alarm generation.

The following are not supported by path selection:

- Packet-mode optimization
- IPv6 optimization
- WCCP designs with Layer-2 redirection
- Designs requiring specific LAN-side redirection
- Layer-2 WAN
- Single-ended SCPS connections
- Maintaining VLAN transparency

For example, network designs in which the in-path interface is sitting on a VLAN trunk connection and you want to switch a flow onto another VLAN, results in discarded packets, because the VLAN ID field is not reflected upon steering.

Path Selection using GRE encapsulation has the following additional restrictions:

- Virtual in-path deployment is not supported.
- Inbound QoS is not applicable to inner or encapsulated incoming traffic.
- Simplified routing does not learn from tunneled packets. If the default gateway is pointed to the WAN, make sure that you have configured the proper static routes for networks that reside in the LAN.
- Flow export or reports reliant on flow show GRE traffic on the WAN interface. Visibility tools that coalesce or stitch LAN and WAN flows together can be affected adversely.
- The downstream Steelhead appliances in serial deployments cannot intercept and take over new TCP connections when an upstream Steelhead appliance sends GRE traffic. Even in the event of admission control, a Steelhead appliance continues to perform path selection and tunneling, preventing proper connection spillover to a downstream Steelhead appliance.

CHAPTER 7 Physical In-Path Deployments

This chapter describes a physical in-path Steelhead appliance deployment. This chapter includes the following sections:

- [“Overview of In-Path Deployment” on page 171](#)
- [“The Logical In-Path Interface” on page 172](#)
- [“Failure Modes” on page 174](#)
- [“Configuring Link State Propagation” on page 176](#)
- [“Cabling and Duplex” on page 177](#)
- [“Physical In-Path Deployment Configuration Examples” on page 180](#)
- [“In-Path Redundancy and Clustering Examples” on page 184](#)
- [“Configuring Simplified Routing” on page 191](#)
- [“Multiple WAN Router Deployments” on page 192](#)
- [“802.1Q Trunk Deployments” on page 206](#)
- [“Layer-2 WAN Deployments” on page 209](#)
- [“VLAN Bridging Deployments” on page 211](#)

This chapter requires that you be familiar with:

- Hot Standby Router Protocol (HSRP)
- Virtual Router Redundancy Protocol (VRRP)
- Gateway Load Balancing Protocol (GLBP)

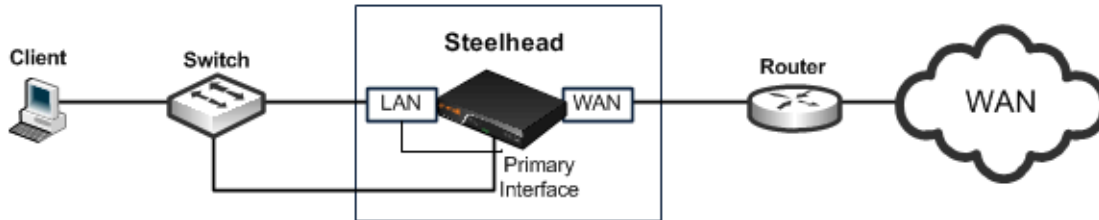
Overview of In-Path Deployment

In a physical in-path Steelhead appliance deployment, a Steelhead appliance LAN interface connects to a LAN-side device (typically a switch), and a corresponding Steelhead appliance WAN interface connects to a WAN connecting device (typically a router). This allows the Steelhead appliance to detect all traffic flowing to and from the WAN and to perform optimization.

Depending on the Steelhead appliance model and its hardware configuration, you can use multiple pairs of WAN and LAN interfaces simultaneously, and you can connect them to multiple switches and routers.

Figure 7-1 shows the simplest type of physical in-path Steelhead appliance deployment.

Figure 7-1. Single Subnet, Physical In-Path Deployment



Most Steelhead appliance deployments are physical in-path deployments. Physical in-path configurations are the easiest to deploy and do not require ongoing maintenance as other configurations do (such as virtual in-path configurations: WCCP, PBR, and Layer-4 redirection).

For networks that contain firewalls or tunnels (VPN, GRE, IPSec transport mode) between Steelhead appliances and require manual tuning of the MTU values, see [“MTU Sizing” on page 445](#).

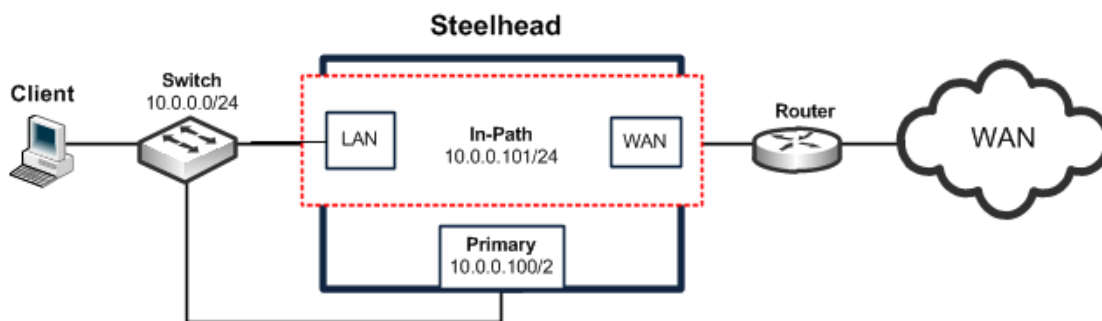
The Logical In-Path Interface

All Steelhead appliances ship with at least one pair of ports that are used for in-path deployments. This pair of ports forms the logical in-path interface. The logical in-path interface acts as an independent, two-port bridge, with its own IP address. This section includes the following topics:

- [“In-Path IP Address Selection” on page 173](#)
- [“In-Path Default Gateway and Routing” on page 173](#)

Figure 7-2 shows the Steelhead appliance logical in-path interface and how it is physically connected to network devices in a single subnet, in-path deployment.

Figure 7-2. The Logical In-Path Interface in a Single Subnet In-Path Deployment



The simplest in-path Steelhead appliance has two IP addresses:

- **Primary** - Used for system management, RiOS data store synchronization, and SNMP.
- **InPath0_0** - Used for optimized data transmission.

Several types of network interface cards (bypass cards) are available for Steelhead appliances. The desktop Steelhead appliances have network bypass functionality built in. With 1U and 3U systems, you can choose the type of bypass card. Steelhead appliances can have both copper and fiber Ethernet bypass cards.

For details on bypass cards, see the *Network Interface Card Installation Guide* on the Riverbed Support site.

In-Path IP Address Selection

An IP address is required for each Steelhead appliance in-path interface. When using correct addressing or port transparency, the IP address must be reachable by remote Steelhead appliances for optimization to occur.

In some environments, the link between the switch and the router might reside in a subnet that has no available IP address. You can use the following solutions to accommodate the IP address requirement:

- creating a secondary interface, with a new subnet and IP address on the router or switch, and pulling the Steelhead appliance in-path interface IP address from the new subnet.
- creating a new 802.1Q VLAN interface and subnet on the router and switch link, and pulling the Steelhead appliance in-path interface IP address from the new subnet. This also requires entering the appropriate in-path VLAN tag on the Steelhead appliance.

With RiOS v5.0.x or later, you can deploy Steelhead appliances so that the in-path interface IP address is not actually used. This deployment option can be useful for integrating with certain network configurations, such as NAT. However, an IP address must be configured for each enabled in-path interface.

For details on correct addressing, port transparency, and full transparency, see [“WAN Visibility Modes” on page 49](#). For more details on deploying a Steelhead appliance into an existing network, see the Riverbed Knowledge Base article *Steelhead appliance Deployment onto an Existing /30 Network* at <https://supportkb.riverbed.com/support/index?page=content&id=S14964>.

In-Path Default Gateway and Routing

Almost all in-path deployments require the configuration of a default gateway for the in-path interfaces. A physical in-path Steelhead appliance might need to transmit packets from its in-path interface to any:

- local hosts, for the LAN-side of any optimized connections.
- remote Steelhead appliances, for the WAN-side of any optimized connections.
- remote hosts, when transmitting packets during auto-discovery.
- local Steelhead and Interceptor appliances, when communicating with connection-forwarding neighbors.

You must configure an in-path gateway if any of these devices is on a different subnet from the in-path interface.

In small branches, where a Steelhead appliance is physically placed between an access switch and a router or firewall, and all hosts are on the same subnet, then the in-path default gateway must use the same IP address that the local hosts use—that of the router or firewall. With this configuration, the Steelhead appliance uses the gateway as the Layer-2 next hop when transmitting to remote hosts or Steelhead appliances, and uses MAC address discovery through ARP when transmitting packets to the local hosts.

In larger branches, where the Steelhead appliance are deployed between two Layer-3 devices (for example, between a Layer-3 switch and a WAN-side router), then the Steelhead appliance can be configured with a specific in-path gateway, static routes, and simplified routing to ensure that it always transmits packets to the optimal next hop. Although it is impossible to generalize for all environments, a typical configuration for locations that minimize packet ricochet and ensure the best performance:

- use the WAN-side Layer-3 device as the in-path default gateway.
- use the simplified routing destination-only option.
- use the enhanced auto-discovery feature.

Some environments require different settings or additional configuration. For more information, see the Riverbed Support site at <https://support.riverbed.com>.

Failure Modes

This section describes the Steelhead appliance failure modes. This section includes the following topics:

- [“Fail-to-Wire Mode” on page 174](#)
- [“Fail-to-Block Mode” on page 175](#)
- [“Configuring Failure Modes” on page 176](#)

All Steelhead appliance models and in-path network interface cards support fail-to-wire mode. In the event of a disk failure, a software crash, a runaway software process, or even loss of power to the Steelhead appliance, the LAN and WAN ports that form the logical in-path interface become internally connected as if they were the ends of a crossover cable, thereby providing uninterrupted transmission of data over the WAN.

Certain in-path network interface cards also support a fail-to-block mode, where in the case of a failure or loss of power, the Steelhead appliance LAN and WAN interfaces completely lose link status, blocking traffic and forcing it to be rerouted onto paths where the remaining Steelhead appliances are deployed. The default failure mode is fail-to-wire mode.

For a list of in-path network interface cards or bypass cards that support fail-to-block mode, see [“Fail-to-Block Mode” on page 175](#).

If a Steelhead appliance transitions to fail-to-wire or fail-to-block mode, you are notified in the following ways:

- The Intercept/Bypass status light is on.
- **Critical** appears in the Management Console status bar.
- SNMP traps are sent (if you have set this option).
- The event is logged to system logs (syslog) (if you have set this option).
- Email notifications are sent (if you have set this option).

Fail-to-Wire Mode

Fail-to-wire mode allows the Steelhead appliance WAN and LAN ports to serve in the same way as an Ethernet crossover cable. In fail-to-wire mode, Steelhead appliances cannot view or optimize traffic. Instead, all traffic is passed through the Steelhead appliance unoptimized.

All Steelhead appliance in-path interfaces support fail-to-wire mode. Fail-to-wire mode is the default setting for Steelhead appliances.

When a Steelhead appliance transitions from normal operation to fail-to-wire mode, Steelhead appliance circuitry physically moves to electronically connect the Steelhead appliance LAN and WAN ports to each other, and physically disconnects these two ports from the rest of the Steelhead appliance. During the transition to fail-to-wire mode, device linked to the Steelhead appliance momentarily disconnect and then immediately connect again. After the transition, traffic resumes flowing as quickly as the connected devices are able to process it. For example, spanning-tree configuration and routing-protocol configuration influence how quickly traffic resumes flowing. Traffic that was passed-through is uninterrupted. Traffic that was optimized might be interrupted, depending on the behavior of the application-layer protocols. When connections are restored, the traffic resumes flowing, although without optimization.

After the Steelhead appliance returns to normal operation, it transitions the Steelhead appliance LAN and WAN ports out of fail-to-wire mode. The devices connected to the Steelhead appliance perceive this as another link state transition. After they are back online, new connections that are made are optimized. However, connections made during the failure are not optimized.

To force all connections to be optimized, you can enable the kickoff feature. This feature resets established connections to force them to go through the connection creation process again. For this reason, before enabling the kickoff feature in production deployments, you must understand and accept that all TCP connections are reset. Generally, connections are short lived and kickoff is not necessary.

For details on enabling the kickoff feature, see [“The Kickoff and Automatic Kickoff Features” on page 26](#) and the *Steelhead Appliance Management Console User’s Guide*.

When a Steelhead appliance transitions to fail-to-wire mode, the transition can have an effect on devices connected to the Steelhead appliance. For example, one common implication pertains to the spanning-tree protocol. In many physical in-path deployments, the Steelhead appliance LAN port is connected to an Ethernet switch, and the Steelhead appliance WAN port is connected to a router.

When a Steelhead appliance transitions from bridging mode to failure mode, a switch might force the port that is connected to the Steelhead appliance to go through the 30-45 second, non-forwarding states of spanning tree. This can result in packet delay or packet loss.

You can resolve this issue by making configuration modifications on your switch. Depending on your switch vendor, there are many different methods to alleviate this issue, ranging from skipping the non-forwarding states (for example, running the **spanning-tree portfast** command on Cisco switches), to using newer 802.1d STP protocols that converge faster on link transitions.

RiOS v5.0.x or later has this mode transition issue only when the Steelhead appliance experiences a power loss. RiOS v4.1 and earlier has this transition state issue when the Steelhead appliance experiences a power loss, software failure, or when the optimization service is restarted.

Fail-to-Block Mode

Some network interfaces support fail-to-block mode. In fail-to-block mode, if the Steelhead appliance has an internal software failure or power loss, the Steelhead appliance LAN and WAN interfaces power down and stop bridging traffic. Fail-to-block mode is useful only if the network has a routing or switching infrastructure that can automatically divert traffic from the link after the failed Steelhead appliance blocks it. You can use fail-to-block mode with connection forwarding, the **allow-failure** CLI command, and an additional Steelhead appliance on another path to the WAN to achieve redundancy.

For more information about connection forwarding and fail-to-block, see [“Configuring Connection Forwarding with Allow-Failure and Fail-to-Block” on page 200](#).

Check the *Network Interface Card Installation Guide* on the Riverbed Support site for a current list of Steelhead appliance in-path interfaces that support fail-to-block mode.

Configuring Failure Modes

This section shows common failure mode configurations using the CLI.

To enable fail-to-block mode

- Connect to the Steelhead CLI and enter the following commands:

```
enable
configure terminal
no interface inpath0_0 fail-to-bypass enable
write memory
```

Note: The changes take effect immediately. You must save your changes or they are lost upon reboot.

To change from fail-to-block mode back to fail-to-wire mode

- Connect to the Steelhead CLI and enter the following commands:

```
enable
configure terminal
interface inpath0_0 fail-to-bypass enable
write memory
```

Note: The changes take effect immediately. You must save your changes or they are lost upon reboot.

To check failure mode status

- Connect to the Steelhead CLI and enter the following commands:

```
enable
show interface inpath0_0
```

Configuring Link State Propagation

In physically in-path deployments, link state propagation (LSP) helps communicate link status between the devices connected to the Steelhead appliance. When this feature is enabled, the link state of each Steelhead appliance LAN and WAN pair is monitored. If either physical port loses link status, the link of the corresponding physical port is also brought down. Link state propagation allows link failure to quickly propagate through the Steelhead appliance, and it is useful in environments where link status is used as a fast-fail trigger.

For example, in a physical in-path deployment where the Steelhead appliance is connected to a router on its WAN port and a switch on its LAN port, if the cable to the router is disconnected, the Steelhead appliance deactivates the link on its LAN port. This causes the switch interface that is connected to the Steelhead appliance to also lose the link. The reverse is also true: if the cable to the switch is disconnected, the router interface that is connected to the Steelhead appliance loses the link.

You can use LSP in a Steelhead appliance serial cluster. In a serial cluster deployment, link state propagation can be useful to quickly propagate failure if the cables between Steelhead appliances are disconnected. For example, in a two-appliance Steelhead appliance serial cluster, if you disconnect the cable between the Steelhead appliances, then both the WAN-side router and the LAN-side switch lose the link.

Link state propagation is supported on either all or none of the interfaces of a Steelhead appliance; it cannot be used to selectively activate an in-path interface.


To enable link state propagation on a Steelhead appliance

Connect to the Steelhead CLI and enter the following commands:

```
enable
configure terminal
in-path lsp enable
write memory
```

Note: The changes take effect immediately. You must save your changes or they are lost upon reboot.

In RiOS v6.0 or later, link state propagation is enabled by default.

 CSH models do not support LSP.

 VSH running RiOS v8.0.3 with ESXi 5.0 and later using a Riverbed NIC card support LSP.

These VSH configurations do not support LSP:

- VSH models running ESX/ESXi 4.0 or 4.1
- VSH models running Microsoft Hyper-V
- VSH models running RiOS v8.0.2 and earlier

Cabling and Duplex

Using the appropriate cables and interface settings for in-path deployments are vital to performance and resiliency. The physical cabling and interface settings that connect the Steelhead appliance to the LAN and WAN equipment (typically a switch and router) at the site must be correct. Duplex mismatches between the Steelhead appliance and equipment connected to it, either during normal operations or during failures when the Steelhead appliance is in fail-to-wire mode, have a significant impact on the performance of all traffic passing through the Steelhead appliance. This section includes the following topics:

- [“Choosing the Correct Cables” on page 177](#)
- [“Duplex Configuration” on page 178](#)
- [“Troubleshooting Cable and Duplex Issues” on page 179](#)

A duplex mismatch or incorrect interface settings might cause optimized performance to be less than non-optimized performance or prevent traffic from flowing—even if the Steelhead appliance is configured properly.

Choosing the Correct Cables

The LAN and WAN ports on the Steelhead bypass cards act like host interfaces during normal operation. During fail-to-wire mode, the LAN and WAN ports act as the ends of a crossover cable. Using the correct cable to connect these ports to other network equipment ensures proper operation during fail-to-wire mode and normal operating conditions.

Riverbed recommends that you do not rely on automatic MDI/MDI-X to automatically sense the cable type. The installation might work when the Steelhead appliance is optimizing traffic, but it might not if the in-path bypass card transitions to fail-to-wire mode.

One way to help ensure that you use the correct cables during an installation is to connect the LAN and WAN interfaces of the Steelhead appliance while the Steelhead appliance is powered off. This proves that the devices either side of the Steelhead appliance can communicate correctly without any errors or other problems.

In the most common in-path configuration, a Steelhead appliance's LAN port is connected to a switch and the Steelhead appliance's WAN port is connected to a router. In this configuration, a straight-through Ethernet cable can connect the Steelhead appliance's LAN to the switch, and a crossover cable must be used to connect the Steelhead appliance's WAN port to the router.

The following table summarizes the correct cable usage in the Steelhead appliance.

Devices	Cable
Steelhead appliance or Interceptor appliance to Steelhead appliance or Interceptor appliance	Crossover
Steelhead appliance or Interceptor appliance to router	Crossover
Steelhead appliance or Interceptor appliance to switch	Straight-through
Steelhead appliance or Interceptor appliance to host	Crossover

Duplex Configuration

Depending on which Steelhead bypass card you use, you must choose between manually setting the speed and duplex for its LAN and WAN interfaces or allowing the interfaces to automatically negotiate. Choosing the correct setting ensures that packets can pass through the interfaces, both during normal operating mode and during fail-to-wire mode, without any errors or drops due to a mismatch between the Steelhead appliance and its connected network equipment.

The correct duplex settings to use depends on the capabilities of all of the interfaces in the *chain of in-path interfaces*: the connected LAN device (typically a switch), the LAN and WAN ports on the Steelhead bypass cards in use by one or more in-path Steelhead appliances, and the connected device (typically a router). A typical in-path deployment has a Steelhead appliance LAN port connecting to a switch port that is 10/100/1000 Mbps capable, but the Steelhead appliance WAN port connects to a router interface that is only capable of 10/100. In this deployment, manually set both the Steelhead appliance LAN and WAN ports to use 100 Mbps, full duplex. This ensures correct operation during normal operation and fail-to-wire mode.

Riverbed recommends the following:

- If all interfaces in the in-path chain are capable of 1 Gbps or higher speeds, use automatic negotiation on all interfaces in the in-path chain.
- Configure for automatic negotiation on all Ethernet ports running at 100Mbps unless you know for sure of a specific automatic negotiation issue between the Steelhead appliance port and the peer device.
- Never use half duplex—either set manually to full duplex or use automatic negotiation.

If you deviate from these recommendations, you must perform tests to verify that traffic flows when the Steelhead appliance is optimizing traffic and entered fail-to-wire mode.

For example, this can happen if the LAN interface is connected to a 1Gbps device, and the WAN interface is connected to a 100Mbps device, and WAN bandwidth close or equal to 100Mbps. To avoid any potential bottleneck that prevents the Steelhead appliance LAN interface from sending or receiving data at a rate greater than 100Mbps, it is prudent for you to use automatic negotiation on both LAN and WAN interfaces. If you use automatic negotiation, the Steelhead appliance can perform at its best both on the LAN side and the WAN side.

Duplex misconfiguration is not limited to the chain of in-path interfaces, especially at remote locations, where WAN limitations restrict the potential performance of applications. There might be long-standing, unrealized duplex-related errors in the existing LAN infrastructure, or even on host interfaces. You might only discover these long-standing issues when a Steelhead appliance is deployed at the site and attention is concentrated on achieving high performance. Due to the infrastructure duplex problems, the Steelhead appliance performance gains are not as significant as expected. Any such infrastructure issues limit the optimization possible by deploying a Steelhead appliance, and must be resolved to realize the full benefits of a Steelhead appliance deployment.

Signs of a duplex misconfiguration include:

- You cannot connect to an attached device.
- You can connect to a device when you choose automatic negotiation, but you cannot connect to that same device when you manually set the speed or duplex.
- You detect performance issues across the network.

Troubleshooting Cable and Duplex Issues

This section shows common cable and duplex troubleshooting procedures.

To verify if slow performance on the network is due to a duplex problem on the chain of in-path interfaces

1. From the Management Console, open Reports > Networking > Interface Counters.
2. Look for positive values for the following fields:
 - Discards
 - Errors
 - Overruns
 - Frame
 - Carrier counts
 - Collisions

These values are zero on a healthy network, unless you use half duplex. Riverbed recommends that you do not use half duplex.

To verify if slow performance on the network is due to a duplex problem within the LAN infrastructure, and not on an interface in the in-path chain

1. From the Management Console, open the Reports > Networking > Current Connections.
2. Look for any optimized connection, and click the magnifying glass icon next to the connection to see its details.

3. Look for zeroes in the following fields:

- Retransmitted
- Fast Retransmitted
- Time-outs

If the values are greater than zero, some type of LAN side packet loss was experienced for that connection. This might be because of a duplex misconfiguration somewhere between the local host and the Steelhead appliance's LAN interface.

Note: Speed and duplex issues might be present at other points in the network path besides the interfaces directly connected to the Steelhead appliance. There might be long-standing interface errors within the LAN, whose symptoms might have been incorrectly blamed on WAN performance.

Physical In-Path Deployment Configuration Examples

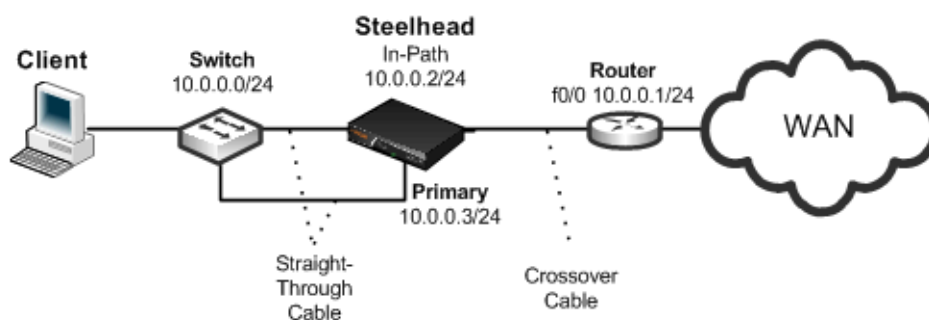
The following section describes common deployment options. This section includes the following topics:

- [“Configuring a Basic Physical In-Path Deployment” on page 180](#)
- [“Configuring a Physical In-Path with Dual Links Deployment” on page 182](#)
- [“Configuring a Serial Cluster Deployment with Multiple Links” on page 183](#)

Configuring a Basic Physical In-Path Deployment

The simplest physical in-path Steelhead appliance deployment is also the most commonly deployed.

Figure 7-3. Simple, Physical In-Path Deployment



Basic steps to perform before you deploy a physical in-path Steelhead appliance

1. Determine the speed for the:
 - switch interface.
 - router interface.

- Steelhead appliance primary interface.
- Steelhead appliance WAN interface.
- Steelhead appliance LAN interface.

Riverbed recommends the following speeds:

- **Fast Ethernet interfaces** - 100 Mb full duplex
- **Gigabit interfaces** - 1000 Mb full duplex

2. Determine the IP addresses for the Steelhead appliance. A Steelhead appliance that is deployed in a physical in-path mode requires two IP addresses, one each for the:

- Steelhead appliance in-path interface.
- Steelhead appliance primary interface (used for managing the Steelhead appliance).

In addition to using the primary interface for management purposes, you can also use the auxiliary (AUX) interface or the in-path management interface to manage the Steelhead appliance. The AUX interface is another physical interface on the Steelhead appliance, and the in-path management interface is a virtual interface that is associated to the Steelhead appliance in-path interfaces.

When you configure the AUX interface, you can not have it in the same subnet as the primary interface.

Each one in-path management interface has one in-path interface. For example, inpath0_0 has a corresponding mgmt0_0 interface, inpath0_1 has a corresponding mgmt0_1 interface, and so on. Any connections destined to the in-path management interface are not optimized, and these connections do not appear in the Current Connections report.

The following are characteristics of the in-path management interface:

- Must be in its own subnet
- Cannot share the same subnet with any other interfaces on the Steelhead appliance (this includes other in-path interfaces)
- Is accessible from either the LAN or WAN side
- Uses the main routing table and is always up
- Supports 802.1Q and processes only packets destined to its VLAN ID (if you configure one)

3. Manually configure the speed for the:

- switch interface.
- router interface.
- Steelhead appliance primary interface.
- Steelhead appliance WAN interface.
- Steelhead appliance LAN interface.

Important: Riverbed strongly recommends that you manually configure interface speed, unless you want to use 1 Gbps, which must be set using automatic negotiation. For details, see <https://supportkb.riverbed.com/support/index?page=content&id=S14623>.

4. Configure the appropriate default gateway for the primary and in-path interfaces:

- **Primary port gateway IP** - Specify the primary gateway IP address.
- **In-path gateway IP** - Specify the IP address for the in-path gateway. If you have a router (or a Layer-3 switch) on the LAN side of your network, specify this device as the in-path gateway. Make sure that you have simplified routing enabled.

For more information, see [“In-Path Default Gateway and Routing” on page 173](#) and [“Configuring a Physical In-Path with Dual Links Deployment” on page 182](#).

Using [Figure 7-3](#) as your environment, the following task includes the minimum steps required to configure the simplest physical in-path Steelhead appliance deployment.

The example requires that you have configured your cabling and duplex according to the recommendations described in [“Cabling and Duplex” on page 177](#).

To configure the Steelhead appliance for basic physical in-path deployment

- On the Steelhead appliance, connect to the CLI and enter the following commands:

```
enable
configure terminal
interface inpath0_0 ip address 10.0.0.2 /24
ip in-path-gateway inpath0_0 10.0.0.1
interface primary ip address 10.0.0.3 /24
ip default-gateway 10.0.0.1
in-path enable
```

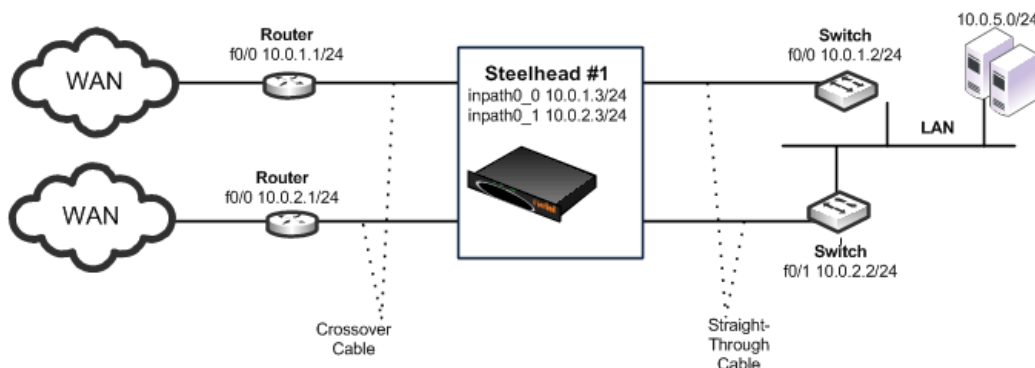
Configuring a Physical In-Path with Dual Links Deployment

This example requires that you have configured your cabling and duplex according to the recommendations described in [“Cabling and Duplex” on page 177](#).

[Figure 7-4](#) shows a physical in-path with dual links Steelhead appliance deployment.

Note: Simplified routing removes any packet ricochet that occurs when the Steelhead appliance sends traffic to the 10.0.5.0/24 LAN.

Figure 7-4. Physical In-Path with Dual Links Deployment



The following Steelhead CLI commands are the minimum commands required to configure the physical in-path Steelhead appliance with dual links. These commands do not include the configuration of features such as duplex, alarms, SNMP, and DNS.

To configure a Steelhead appliance physically in-path with dual links

1. On Steelhead 1, connect to the CLI and enter the following commands:

```
enable
configure terminal
interface inpath0_0 ip address 10.0.1.3 /24
ip in-path-gateway inpath0_0 10.0.1.2
interface inpath0_1 ip address 10.0.2.3 /24
ip in-path-gateway inpath0_1 10.0.2.2
in-path enable
in-path peering auto
in-path simplified routing all
write memory
restart
```

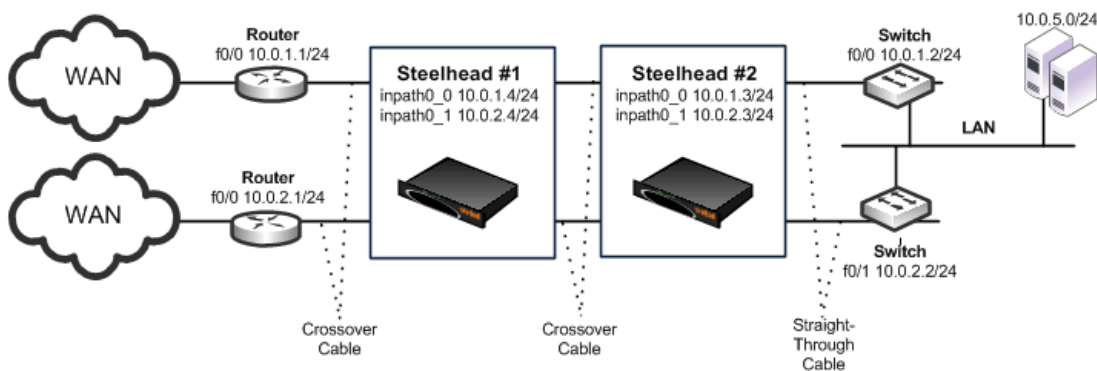
Configuring a Serial Cluster Deployment with Multiple Links

This example requires that you have configured your cabling and duplex according to the recommendations described in [“Cabling and Duplex” on page 177](#).

[Figure 7-5](#) shows a serial cluster deployment with multiple WAN links. Each of the links are on different subnets, but they might also be in the same subnet.

Note: Link state propagation is enabled between the Steelhead appliances. For details, see *Steelhead Appliance Management Console User's Guide*.

Figure 7-5. Physical In-Path, Multiple Link Serial Cluster Deployment



The following Steelhead CLI commands are the minimum commands required to configure a serially clustered Steelhead appliance deployment with multiple WAN links. These commands do not include the configuration of features such as duplex, alarms, and DNS.

To configure serially clustered Steelhead appliances with multiple WAN links

1. On Steelhead 1, connect to the CLI and enter the following commands:

```
enable
configure terminal
interface inpath0_0 ip address 10.0.1.4 /24
ip in-path-gateway inpath0_0 10.0.1.2
interface inpath0_1 ip address 10.0.2.4 /24
ip in-path-gateway inpath0_1 10.0.2.2
in-path enable
in-path peering auto
in-path simplified routing dest-only
in-path peering rule pass peer 10.0.1.3 rulenum end
in-path peering rule pass peer 10.0.2.3 rulenum end
write memory
restart
```

2. On Steelhead 2, connect to the CLI and enter the following commands:

```
enable
configure terminal
interface inpath0_0 ip address 10.0.1.3 /24
ip in-path-gateway inpath0_0 10.0.1.2
interface inpath0_1 ip address 10.0.2.3 /24
ip in-path-gateway inpath0_1 10.0.2.2
in-path enable
in-path simplified routing dest-only
in-path peering auto
in-path peering rule pass peer 10.0.1.4 rulenum end
in-path peering rule pass peer 10.0.2.4 rulenum end
write memory
restart
```

In-Path Redundancy and Clustering Examples

You can use the following techniques to configure multiple Steelhead appliances in physical in-path deployments. These deployments achieve redundancy and clustering for optimization. This section covers the following scenarios:

- [“Configuring Master and Backup Deployments” on page 184](#)
- [“Configuring Serial Cluster Deployments” on page 187](#)

You can use the techniques in each scenario to provide optimization across several physical links. You can use these techniques in conjunction with connection forwarding when all of the physical links to and from the WAN are unable to pass through a single Steelhead appliance.

For details on connection forwarding, see [“Connection Forwarding” on page 42](#).

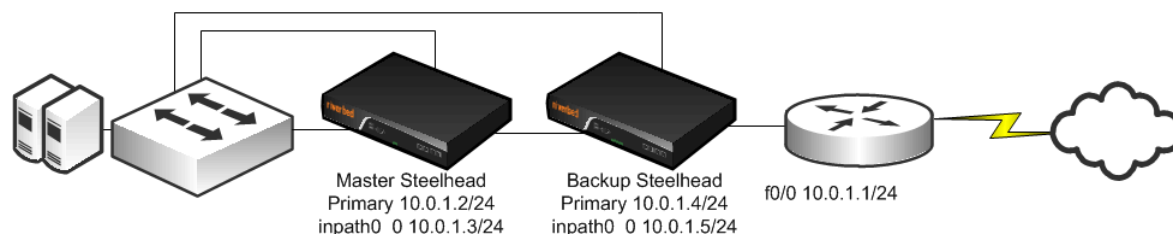
Configuring Master and Backup Deployments

In a master and backup deployment, two equivalent model Steelhead appliances are placed physically in-path. This section includes the following topics:

- [“Configuring a Master and Backup Deployment” on page 186](#)
- [“Adjusting the Timers for Faster Master and Backup Failover” on page 186](#)

The Steelhead appliance closest to the LAN is configured as a master, and the other Steelhead appliance as the backup. The master Steelhead appliance optimizes traffic and the backup Steelhead appliance checks to make sure the master Steelhead appliance is functioning and not in *admission control*. Admission control means the Steelhead appliance has stopped trying to optimize new connections, due to reaching its TCP connection limit, or due to some abnormal condition. If the backup Steelhead appliance cannot reach the master, or if the master has entered admission control, the backup Steelhead appliance begins optimizing new connections until the master recovers. After the master has recovered, the backup Steelhead appliance stops optimizing new connections, but continues to optimize any existing connections that were made while the master was down. The recovered master optimizes any newly formed connections.

Figure 7-6. Master and Backup Deployment



Riverbed recommends that you do not use the master and backup deployment method on Steelhead appliances with multiple in-path interfaces, because only a single in-path interface communicates between the master and a backup. By default, the `inpath0_0` interface is used, or the lowest numbered active in-path interface. The master and backup appliances create a TCP connection to each other sourced from this interface, even if other interfaces are available. If the Steelhead appliances cannot reach each other through this interface, they do not attempt to use other interfaces to establish connectivity. Instead, the Steelhead appliance behaves as if the other Steelhead appliance is down or in admission control. If the `inpath0_0` interface connection breaks but both Steelhead appliances are running, both appliances consider themselves to be the master and both appliances optimize connections.

If you use the master and backup deployment method with Steelhead appliances that have multiple active in-path links, peering rules must also be configured. Add peering rules to both Steelhead appliances for each in-path interface; these peering rules must have an action *pass* for a peer IP address of each of the in-path IP addresses. This setting ensures that during any window of time during which both Steelhead appliances are active (for example, during a master recovery), the Steelhead appliances do not try to optimize connections between themselves.

Typically, RiOS data store synchronization is used in master and backup deployments. RiOS data store synchronization ensures that any data written to one Steelhead appliance eventually is pushed to the other Steelhead appliance. Although both the master and backup deployment option and the RiOS data store synchronization feature use the terms *master* and *backup*, the uses are different and separate. You can typically configure one Steelhead appliance to be a master for both, but it is not a requirement.

For details on data synchronization, see [“RiOS Data Store Synchronization” on page 15](#).

Consider using a master and backup deployment instead of a serial cluster when all of the following are true:

- Only two Steelhead appliances are placed physically in-path.
- The capacity of a single Steelhead appliance is sufficient for the site.
- Only a single in-path interface is active on both Steelhead appliances.

Some environments might require additional considerations. For more information, see the Riverbed Support site at <https://support.riverbed.com>.

Configuring a Master and Backup Deployment

This section describes how to configure the master and backup deployment shown in [Figure 7-6](#).

To configure the master and backup Steelhead appliances

1. Connect to the master Steelhead CLI and enter the following commands:

```
# -- Master Steelhead appliance
interface primary ip address 10.0.1.2/24
ip default gateway 10.0.1.1
interface inpath0_0 ip address 10.0.1.3/24
ip in-path-gateway inpath0_0 10.0.1.1
# -- Failover should point to the inpath0_0 address.
failover steelhead addr 10.0.1.5
failover master
failover enable
in-path enable
# -- Although not required, RiOS data store synchronization is usually enabled
# -- in master/backup deployments
datastore sync master
# -- RiOS data store should point to peer primary or aux interface address
datastore sync peer-ip 10.0.1.4
datastore sync enable
write memory
restart
```

2. Connect to the backup Steelhead CLI and enter the following commands:

```
# -- Backup Steelhead appliance
interface primary ip address 10.0.1.4/24
ip default gateway 10.0.1.1
interface inpath0_0 ip address 10.0.1.5/24
ip in-path-gateway inpath0_0 10.0.1.1
# -- Failover should point to the inpath0_0 address.
failover steelhead addr 10.0.1.3
no failover master
failover enable
in-path enable
# -- Although not required, RiOS data store synchronization is usually enabled in
# -- master/backup deployments
no datastore sync master
# -- RiOS data store should point to peer's primary or aux interface address.
datastore sync peer-ip 10.0.1.2
datastore sync enable
write memory
restart
```

Note: For more details on configuring master and backup deployment, see the Failover Support commands in the *Riverbed Command-Line Interface Reference Manual*, and the Enabling Failover section in the *Steelhead Appliance Management Console User's Guide*.

Adjusting the Timers for Faster Master and Backup Failover

In a steady, normal operating state, the backup Steelhead appliance periodically sends keep-alive messages to the master Steelhead appliance on TCP port 7820. If the master Steelhead appliance does not respond to the keep-alive message within five seconds, the backup Steelhead appliance drops the connection and attempts to reconnect to the master Steelhead appliance. The backup Steelhead appliance attempts to reconnect a maximum of five times, and each time it waits for two seconds before aborting the connection.

If all connection attempts fail, the backup Steelhead appliance transitions into an active state and starts optimizing the connections. If you use the default value failover settings, it can take as long as 15 seconds before the backup Steelhead appliance starts optimizing connections.

You can adjust several failover settings to shorten the failover time:

- **Read timeout (in milliseconds)** - Governs how many milliseconds the backup Steelhead appliance waits for the master Steelhead appliance to respond to its keep-alive messages. Use the **failover read timeout** command to adjust this setting. The default value is 5000 ms.
- **Connection attempts** - The number of times the backup Steelhead appliance attempts to reconnect to the master Steelhead appliance after read time-out has expired. Use the **failover connection attempts** command to adjust this setting. The default value is 5.
- **Connection timeout (in milliseconds)** - The number of milliseconds the backup Steelhead appliance waits before aborting the reconnection attempt to the master Steelhead appliance. Use the **failover connection timeout** command to adjust this setting. The default value is 2000 ms.

To reduce the failover time to five seconds, you can adjust the timers to the following:

- **Read timeout** - 1000 ms
- **Connection attempts** - 4
- **Connection timeout** - 1000 ms

Configuring Serial Cluster Deployments

You can provide increased optimization by deploying two or more Steelhead appliances back-to-back in an in-path configuration to create a serial cluster. This section includes the following topics:

- [“Serial Cluster Rules” on page 188](#)
- [“Configuring a Basic Serial Cluster Deployment” on page 188](#)
- [“Configuring Faster Peer Failure Detection” on page 190](#)

Steelhead appliances in a serial cluster process the peering rules you specify in a spillover fashion. When the maximum number of TCP connections for a Steelhead appliance is reached, that appliance stops intercepting new connections. This allows the next Steelhead appliance in the cluster the opportunity to intercept the new connection, if it has not reached its maximum number of connections.

The in-path peering rules and in-path rules tell the Steelhead appliance in a cluster not to intercept connections between themselves. You configure peering rules that define what to do when a Steelhead appliance receives an auto-discovery probe from another Steelhead appliance. You can deploy serial clusters on the client or server-side of the network.

Important: For environments in which you want to optimize MAPI or FTP traffic, which require all connections from a client to be optimized by one Steelhead appliance, Riverbed strongly recommends using the master and backup redundancy configuration instead of a serial cluster deployment. For larger environments that require multiple appliance scalability and high availability, Riverbed recommends using the Interceptor appliance to build multiple appliance clusters. For details, see the *Interceptor Appliance User's Guide*.

Before you configure a serial cluster deployment, consider the following factors:

- The total optimized WAN capacity of the cluster can reach the sum of the optimized WAN capacity of the individual Steelhead appliances, assuming that both Steelhead appliances are optimizing connections. Typically, both Steelhead appliances optimize connections if connections originate from both WAN and LAN, or if one of the Steelhead appliances reaches its capacity limit and passes through subsequent connections (which are then optimized by the other Steelhead appliance).
- If the active Steelhead appliance in the cluster enters a degraded state because the CPU load is too high, it continues to accept new connections.

For more details on working with serial clusters, see the Riverbed Knowledge Base article *Working with Serial Clustering* at <https://supportkb.riverbed.com/support/index?page=content&id=s15555>.

Serial Cluster Rules

The in-path peering rules and in-path pass-through rules tell the Steelhead appliances in a serial cluster not to intercept connections between each other. The peering rules define what happens when a Steelhead appliance receives an auto-discovery probe from another Steelhead appliance in the same cluster.

You can deploy serial clusters on the client or server side of the network.

Figure 7-7. Serial Cluster Deployment



In this example, Steelhead 1, Steelhead 2, and Steelhead 3 are configured with in-path peering rules so they do not answer probe requests from one another, and with in-path rules so they do not accept their own WAN connections. Similarly, Steelhead 4, Steelhead 5, and Steelhead 6 are configured so that they do not answer probes from one another and do not intercept inner connections from one another. The Steelhead appliances are configured to find an available peer Steelhead appliance on the other side of the WAN.

Configuring a Basic Serial Cluster Deployment

Figure 7-8 shows an example serial cluster deployment of three in-path Steelhead appliances in a data center.

Figure 7-8. Serial Cluster in a Data Center



This example uses the following parameters:

- Steelhead 1 in-path IP address is 10.0.1.1
- Steelhead 2 in-path IP address is 10.0.1.2
- Steelhead 3 in-path IP address is 10.0.1.3

In this example, you configure each Steelhead appliance with in-path peering rules to prevent peering with another Steelhead appliance in the cluster, and with in-path rules to not optimize connections originating from other Steelhead appliances in the same cluster.

To configure a basic serial cluster deployment with three Steelhead appliances

1. On Steelhead 1, connect to the CLI and enter the following commands:

```
enable
configure terminal
in-path peering rule pass peer 10.0.1.2 rulenum 1
in-path peering rule pass peer 10.0.1.3 rulenum 1
in-path rule pass-through srcaddr 10.0.1.2/32 rulenum 1
in-path rule pass-through srcaddr 10.0.1.3/32 rulenum 1
write memory
show in-path peering rules
```

Rule	Type	Source Network	Dest Network	Port	Peer Addr
1	pass	*	*	*	10.0.1.3
2	pass	*	*	*	10.0.1.2
def auto	*	*	*	*	*

```
show in-path rules
```

Rule	Type	Source Addr	Dest Addr	Port	Target Addr	Port
1	pass	10.0.1.3/24	*	*	--	--
2	pass	10.0.1.2/24	*	*	--	--
def auto	*	*	*	*	--	--

The changes take effect immediately. You must save your changes or they are lost upon reboot.

2. On Steelhead 2, connect to the CLI and enter the following commands:

```
enable
configure terminal
in-path peering rule pass peer 10.0.1.1 rulenum 1
in-path peering rule pass peer 10.0.1.3 rulenum 1
in-path rule pass-through srcaddr 10.0.1.1/32 rulenum 1
in-path rule pass-through srcaddr 10.0.1.3/32 rulenum 1
write memory

show in-path peering rules
```

Rule	Type	Source Network	Dest Network	Port	Peer Addr
1	pass	*	*	*	10.0.1.3
2	pass	*	*	*	10.0.1.1
def auto	*	*	*	*	*

```
show in-path rules
```

Rule	Type	Source Addr	Dest Addr	Port	Target Addr	Port
1	pass	10.0.1.3/24	*	*	--	--
2	pass	10.0.1.1/24	*	*	--	--
def auto	*	*	*	*	--	--

The changes take effect immediately. You must save your changes or they are lost upon reboot.

3. On Steelhead 3, connect to the CLI and enter the following commands:

```
enable
configure terminal
in-path peering rule pass peer 10.0.1.1 rulenum 1
in-path peering rule pass peer 10.0.1.2 rulenum 1
in-path rule pass-through srcaddr 10.0.1.1/32 rulenum 1
in-path rule pass-through srcaddr 10.0.1.2/32 rulenum 1
write memory
```

```
show in-path peering rules
```

Rule	Type	Source Network	Dest Network	Port	Peer Addr
1	pass	*	*	*	10.0.1.2
2	pass	*	*	*	10.0.1.1
def auto	*	*	*	*	*

```
show in-path rules
```

Rule	Type	Source Addr	Dest Addr	Port	Target Addr	Port
1	pass	10.0.1.2/24	*	*	--	--
2	pass	10.0.1.1/24	*	*	--	--
def auto	*	*	*	*	--	--

The changes take effect immediately. You must save your changes or they are lost upon reboot.

Port 7800 is the default pass-through port. The Steelhead appliances by default pass through and not intercept SYN packets arriving on port 7800. These in-path pass-through rules are necessary only if the Steelhead appliances have been configured to use service ports other than 7800 for the Steelhead appliance-to-Steelhead appliance connections.

Configuring Faster Peer Failure Detection

A Steelhead appliance uses the out-of-band (OOB) connection to inform a peer Steelhead appliance of its capabilities. The OOB connection is also used to detect failures. By default, a Steelhead appliance sends a keep-alive message every 20 seconds, and it declares a peer down after sending two keep-alive messages (40 seconds) and no response is received. If you want faster peer failure detection, use the following commands to adjust the interval and the number of keep-alive messages sent:

```
protocol connection wan keep-alive oob def-count (default of 2; minimum value of 2)
protocol connection wan keep-alive oob def-intvl (default of 20; minimum value of 5)
```

Losing the OOB connection does not affect the optimized sessions, because the optimized sessions have a one-to-one mapping between the outer channel (the LAN-side TCP connection between the client and server, and the Steelhead appliance) and the inner channel (the WAN-side TCP connection between the Steelhead appliances). The disadvantage to this approach is that the application does not notice when the peer is unavailable and the application might appear as if it is not working to the end user.

To address this, you can disconnect the inner and outer channels when the Steelhead appliance loses its OOB connection by using the **protocol connection lan on-oob-timeout drop all enable** command.

For Steelhead appliances with multiple in-path interfaces, the **protocol connection lan on-oob-timeout drop all enable** command disconnects all the optimized sessions, even if there are other OOB connections originating from other in-path interfaces. To configure the Steelhead appliance to drop only the connections related to a specific in-path interface, use the command **protocol connection lan on-oob-timeout drop same-inpath enable**.

For more details on OOB, see [“The Out-of-Band Connection” on page 69](#).

Configuring Simplified Routing

Simplified routing is only effective in topologies where the in-path Steelhead appliance resides on a different network than the end hosts. Riverbed recommends that you use simplified routing in deployments where a Layer-3 switch separates the end hosts from the Steelhead appliance.

For an overview of simplified routing and packet ricochet, see [“Overview of Simplified Routing” on page 45](#).

You can enable simplified routing with the **in-path simplified routing <option>** command or in the Management Console. The following options determine if simplified routing is enabled and what packet elements the Steelhead appliance can use (that is, what is *learned*).

To configure simplified routing from the CLI

- Connect to the CLI and enter the following command:

```
in-path simplified routing <option>
```

The following table summarizes the simplified routing command options.

Parameter	Definition
None	Does not collect mappings. This setting disables simplified routing learning. The <none> option is required for virtual in-path deployments.
Destination only	Collects mappings from destination IP, destination MAC, and VLAN tag (when deployed on 802.1q trunk). Riverbed recommends that you use <destination only> for most deployments with multiple in-paths or connection forwarding Steelhead appliances. Destination only is enabled by default on appliances manufactured with RiOS v6.0 or later. Steelhead appliances do not usually learn incorrect mappings unless the network devices themselves are routing incorrectly.
Destination and source	Collects mappings from destination and source IP, destination and source MAC, and VLAN tag (when deployed on 802.1q trunk).
All	Collects mappings for destination and source IP, destination and source MAC, VLAN tag, and Steelhead appliance inner connection traffic and auto-discovery options. This option has the advantage of learning simplified entries faster than the destination only. option Riverbed recommends that you use <all> in topologies when you deploy the Steelhead appliance a 802.1q trunk.

You can view the simplified routing table that includes the in-path interface learned information, the entry, IP address, MAC address, VLAN tag ID, and the times the entry was used.

To view the simplified routing table on the Steelhead appliance

- Connect to the CLI and enter the following command to see the following output:

```
show in-path macmap-tables
relay      ip_addr      mac_addr      vlan  ref_count
inpath0_0  10.18.4.9      00:0d:66:95:e8:00  0      6
```

You can use the output of this command if an entry points to an incorrect MAC address, such as a firewall. Riverbed recommends that you collaborate with Riverbed Support for troubleshooting—the understanding of simplified routing learning in complex topologies can require detailed traffic flow analysis.

Simplified routing has the following constraints:

- The **<none>** option must be used when you configure simplified routing in a virtual in-path (WCCP/PBR/Interceptor appliance) environment.
- The default route must exist on the Steelhead appliance.

In the following deployment examples, the recommended simplified routing settings are specified in the CLI configurations.

Multiple WAN Router Deployments

Typically, multiple WAN routers are used at locations where redundancy or high availability is important. With multiple routers, the loss of a single WAN link, or a single WAN router, does not prevent hosts at the locations from reaching WAN resources. Steelhead appliances can be deployed and configured to maintain the high availability for network access. Additionally, multiple Steelhead appliances can be deployed and configured so that a Steelhead appliance failure allows new connections to be optimized.

This section covers the following sections:

- [“Configuring Multiple WAN Router Deployments Without Connection Forwarding” on page 194](#)
- [“Configuring Multiple WAN Router Deployments with Connection Forwarding” on page 198](#)

If one or more Steelhead appliances are deployed to cover all the links between the LAN switches and the WAN connecting routers, connection forwarding is not required. These deployments are referred to as *serial* deployments, and they can use multiple Steelhead appliances (in a [“Configuring Master and Backup Deployments” on page 184](#) or [“Configuring Serial Cluster Deployments” on page 187](#)) to achieve optimization high availability.

If it is impossible or impractical to have all the WAN links covered by a single Steelhead appliance, multiple Steelhead appliances are used. They must have connection forwarding configured. These deployments are known as *parallel* deployments. High availability for optimization is achieved by using either the connection forwarding fail-to-block configuration, or by using master and backup, or serial clustering on each of the parallel links to the WAN.

For more details on connection forwarding, see [“Connection Forwarding” on page 42](#). For more details on fail-to-block mode, see [“Fail-to-Block Mode” on page 175](#).

Riverbed recommends that you use designs that do not require connection forwarding (that is, serial designs) whenever possible. Serial designs require less configuration, and are easier to troubleshoot, than parallel designs. If you need a parallel design, a deployment using the Interceptor appliance might have several advantages, including policy-based load-balancing and failover handling.

Using the WAN- or LAN-side HSRP IP address improves the likelihood that optimized connections survive a network outage. To plan this you must understand how the Steelhead appliance learns and reacts to changes in the network. Keep in mind that there are many different network design possibilities, and it is not possible to explain all the caveats here.

By default, the Steelhead appliance uses simplified routing, which learns the association between IP addresses and MAC addresses, and Address Resolution Protocol (ARP), which learns associations between IP addresses and MAC addresses on the same subnet as the in-path interface and MAC addresses.

The Steelhead appliance also builds a table of MAC addresses to the LAN or WAN interface, based on the Ethernet frames that cross through the Steelhead appliance in-path interface. Using these tables, the Steelhead appliance learns which destination MAC address to use for packets the Steelhead appliance originates and the corresponding interface on which to transmit the packet. For example, if local hosts are on the same subnet as the Steelhead appliance in-path interfaces and the WAN routers are using HSRP, the transmitting in-path interface learns all local IP addresses on the same subnet to MAC address, through ARP, through the LAN interface. The Steelhead appliance learns remote IP address-to-MAC address relationships from simplified routing = destination-only through the WAN interface. If the Steelhead appliance has not learned the IP address-to-MAC address relationship through simplified routing or ARP, it follows its default gateway.

When the network experiences an outage, the Steelhead appliance in-path interface that transmits for the connection does not react to the change in the network until there has been a change in the flow of traffic through the transmitting in-path interface. After the flow of traffic changes, the Steelhead appliance in-path interface can learn that the destination is available through the opposite interface (WAN now goes to LAN). For example, using the default simplified routing setting, a Steelhead appliance learns that a remote IP address is associated with the MAC address of the primary WAN router that owns the HSRP virtual MAC. When the primary router WAN circuit fails, the primary router can decrement its HSRP priority and the standby router can preempt the primary router to assume control of the HSRP virtual MAC. When this happens, the transmitting interface learns the HSRP MAC through the LAN interface and can continue transmitting traffic across the WAN for optimized connections.

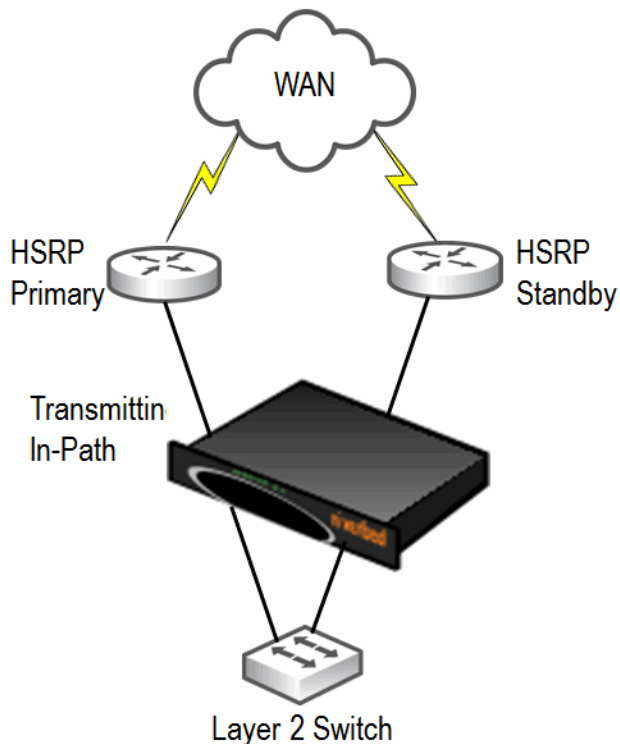
On the other hand, if the transmitting interface learned the remote IP-to-MAC relationship as the physical interface of the primary router then it needs to detect a packet ricochet to learn that the path across the WAN is actually through the LAN interface. Also, you might need to consider any TCP connections that the Steelhead appliance originates, such as the OOB splice. You can use the **protocol connection lan on-oob-timeout drop** command.

In certain scenarios, existing optimized connections do not survive: for example, if there is a failure between the Steelhead appliance and a directly connected device, such as the cable is damaged or a device lost power. When a directly-connected device fails, the first-hop redundancy protocol such as HSRP detects the failure and a standby device can assume the primary role. However, some features on the Steelhead appliance, such as link state propagation, can also detect the failure and stop connectivity in the associated interface (if the failure affects LAN0_0 then WAN0_0 also stops connectivity). The result is that all paths from the in-path interface that transmit for the optimized connection are in a down state and the optimized connections do not continue. Most applications restart new connections. Link state propagation provides feedback to other devices of the failure, and network protocols can more quickly detect the failure.

The choice of a default gateway is very important in locations with multiple WAN routers. In addition to choosing a default gateway (and simplified routing) that minimizes packet ricochet, HSRP or similar protocols can be used to ensure the loss of a single WAN router does not prevent the Steelhead appliance from transmitting packets over the WAN. Most WAN devices that support HSRP or similar protocols have a *link tracking* option that allows them to relinquish the HSRP virtual IP address if a WAN link fails; this option should be used when possible.

Note: In a high-availability environment, there are often multiple gateways or next hops to choose from. To minimize the disruption to any existing optimized connections when a network device fails, it is important that the correct settings are configured on the Steelhead appliances.

Figure 7-9. HSRP Diagram



Configuring Multiple WAN Router Deployments Without Connection Forwarding

The following section describes best practices for serial Steelhead appliance deployments at locations with multiple routers. Each of the following scenarios can be modified to use multiple Steelhead appliances, either in master and backup or serial cluster configurations. This section covers the following scenarios:

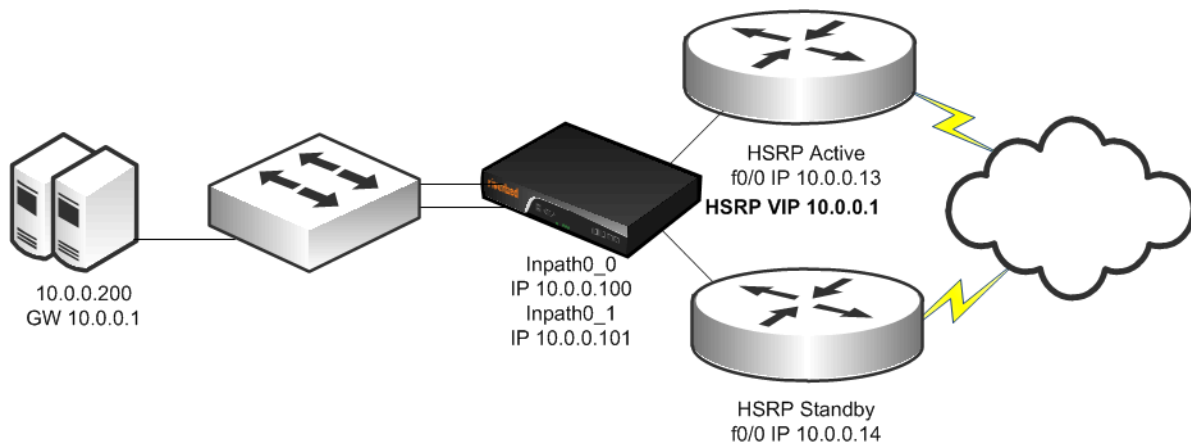
- [“Configuring a Single Steelhead Appliance and Single Layer-2 Switch Deployment” on page 195](#)
- [“Configuring a Single Steelhead Appliance and Dual Layer-2 Switches Deployment” on page 195](#)
- [“Configuring a Single Steelhead Appliance and Single Layer-3 Switch Deployment” on page 196](#)
- [“Configuring Single Steelhead Appliance and Dual Layer-3 Switches Deployment” on page 197](#)

Configuring a Single Steelhead Appliance and Single Layer-2 Switch Deployment

Figure 7-10 shows a topology consisting of two routers, a single Layer-2 switch, and one Steelhead appliance with a 4-port card. The client and the Steelhead appliance are in the same subnet. The client uses the HSRP virtual IP as its default gateway (10.0.0.1).

In this environment, the in-path gateway for both the inpath0_0 and inpath0_1 interfaces must point to the HSRP virtual IP (10.0.0.1). You do not need to enable simplified routing as the client is on the same subnet as the Steelhead appliance.

Figure 7-10. Single Steelhead Appliance, Single Layer-2 Switch, Dual Router Deployment



To configure a Steelhead appliance, single Layer-2 switch, and dual routers

- Connect to the CLI and enter the following commands:

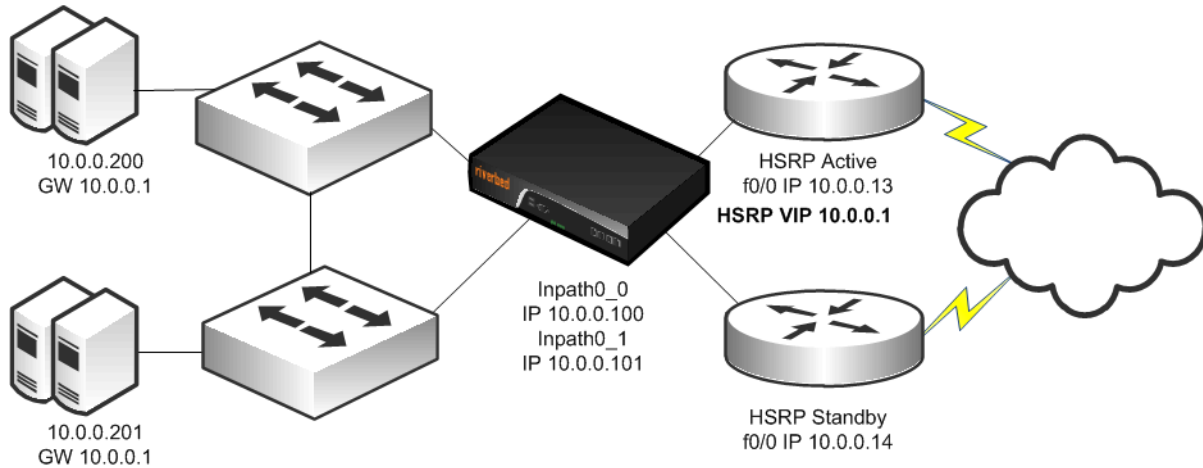
```
#--- Enable and configure the in-path Interfaces
in-path enable
in-path interface inpath0_1 enable
interface inpath0_0 ip address 10.0.0.100 /24
interface inpath0_1 ip address 10.0.0.101 /24
#--- Set the default gateway for the in-path interfaces to be the WAN
#--- side HSRP VIP
ip in-path-gateway inpath0_0 "10.0.0.1"
ip in-path-gateway inpath0_1 "10.0.0.1"
#--- Enable Enhanced auto-discovery, enabled by default on new factory installs of
#--- RiOS v6.x. If you have upgraded a Steelhead appliance to that level or more recently, you
#--- will need to apply this command
in-path peering auto
#--- Simplified Routing is not required but can be enabled. (Simplified Routing
#--- destination only is on by default with new RiOS v6.x installs)
in-path simplified routing dest-only
```

Configuring a Single Steelhead Appliance and Dual Layer-2 Switches Deployment

Figure 7-11 shows a topology in which there are two routers, two Layer-2 switches, and one Steelhead appliance with a 4-port card. The client and the Steelhead appliance are in the same subnet. The client uses the HSRP virtual IP as its default gateway (10.0.0.1).

In this environment, the in-path gateway for both the inpath0_0 and inpath0_1 interfaces must point to the HSRP virtual IP (10.0.0.1). You do not need to enable simplified routing because the clients are on the same subnet as the Steelhead appliance.

Figure 7-11. Single Steelhead Appliance, Dual Layer-2 Switches, Dual Router Deployment



To configure a Steelhead appliance, dual Layer-2 switches, and dual routers

- Connect to the CLI and enter the following commands:

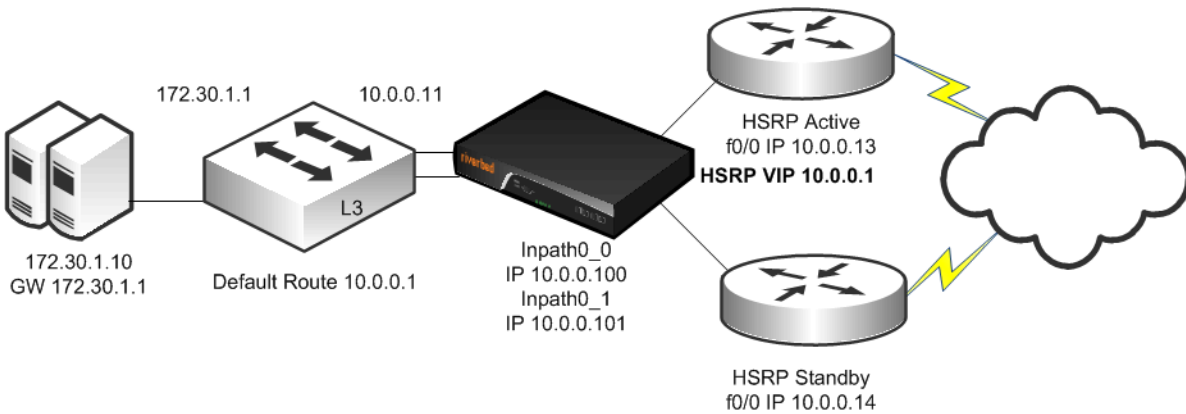
```
#--- Enable and configure the in-path Interfaces
in-path enable
in-path interface inpath0_1 enable
interface inpath0_0 ip address 10.0.0.100 /24
interface inpath0_1 ip address 10.0.0.101 /24
#--- Set the default gateway for the in-path interfaces to be the WAN
#--- side HSRP VIP
ip in-path-gateway inpath0_0 "10.0.0.1"
ip in-path-gateway inpath0_1 "10.0.0.1"
#--- Enable Enhanced auto-discovery, enabled by default on new factory installs of
#--- RiOS v6.x. If you have upgraded a Steelhead appliance to that level or more recently, you
#--- will need to apply this command
in-path peering auto
#--- Simplified Routing is not required but can be enabled. (Simplified Routing
#--- destination only is on by default with new RiOS v6.x installs)
in-path simplified routing dest-only
```

Configuring a Single Steelhead Appliance and Single Layer-3 Switch Deployment

Figure 7-12 shows a topology in which there are two routers, a single Layer-3 switch, and a single Steelhead appliance with a 4-port card. The client and the Steelhead appliance are in different subnets. The client is using the Layer-3 switch as its default gateway. The Layer-3 switch does not have any routing protocols configured and relies on the default route to reach other subnets. The default route uses the HSRP IP address as the next hop.

In this environment, the in-path gateway on the inpath0_0 and inpath0_1 interface must use the Layer-3 switch as its default gateway (10.0.0.11) while configuring simplified routing to populate its table based on destination MAC address (CLI command: **in-path simplified routing dest-only**).

Figure 7-12. Single Steelhead Appliance, Single Layer-3 Switch, Static Routing, Dual Router Deployment



To configure a Steelhead appliance, single Layer-3 switch, static routing, and dual routers

- Connect to the CLI and enter the following commands:

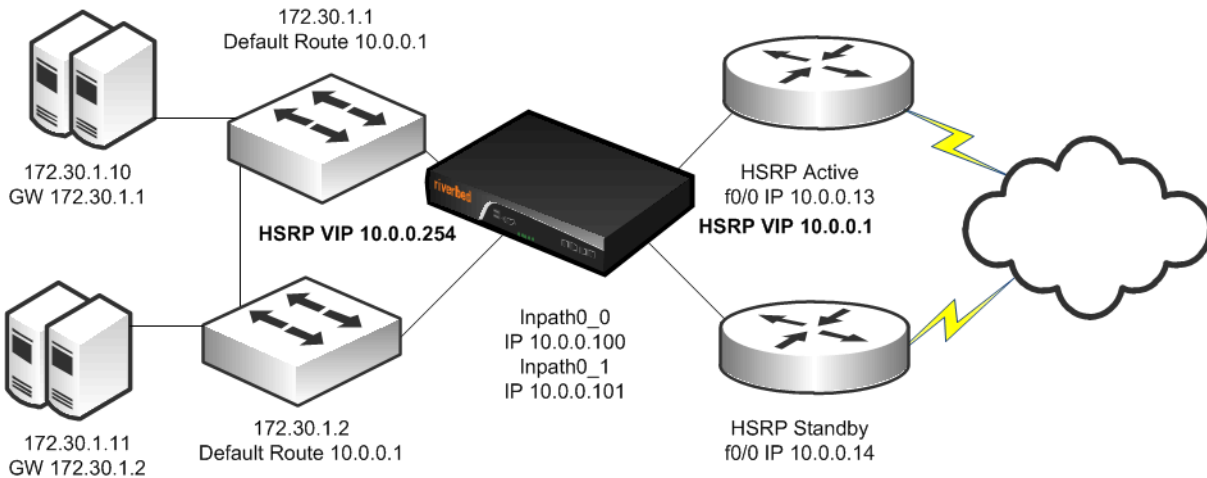
```
#--- Enable and configure the in-path Interfaces
in-path enable
in-path interface inpath0_1 enable
interface inpath0_0 ip address 10.0.0.100 /24
interface inpath0_1 ip address 10.0.0.101 /24
#--- Set the default gateway for the in-path interfaces to be the LAN
#--- side Layer-3 Switch IP
ip in-path-gateway inpath0_0 "10.0.0.11"
ip in-path-gateway inpath0_1 "10.0.0.11"
#--- Enable Enhanced auto-discovery, enabled by default on new factory installs of
#--- RiOS v6.x. If you have upgraded a Steelhead appliance to that level or more recently, you
#--- will need to apply this command
in-path peering auto
#--- Simplified Routing destination only should be used and
#--- is on by default with new RiOS v6.x installs
in-path simplified routing dest-only
```

Configuring Single Steelhead Appliance and Dual Layer-3 Switches Deployment

Figure 7-13 shows a topology in which there are two routers, two Layer-3 switches, and a single Steelhead appliance with a 4-port card. The clients and the Steelhead appliance are in different subnets. The clients are using the Layer-3 switches as their default gateways. The Layer-3 switches do not have any routing protocols configured and relies on the default route to reach other subnets. The default route uses the HSRP IP address as the next hop.

In this environment, the in-path gateway on the inpath0_0 and inpath0_1 interface must use the HSRP address of the Layer-3 switches as its default gateway (10.0.0.254) while configuring simplified routing to populate its table based on destination MAC address (CLI command: **in-path simplified routing dest-only**).

Figure 7-13. Single Steelhead Appliance, Dual Layer-3 Switches, Dual HSRP, Static Routing, Dual Router Deployment



To configure a Steelhead appliance, dual Layer-3 switches, dual HSRP, static routing, and dual routers

Connect to the CLI and enter the following commands:

```
#--- Enable and configure the in-path Interfaces
in-path enable
in-path interface inpath0_1 enable
interface inpath0_0 ip address 10.0.0.100 /24
interface inpath0_1 ip address 10.0.0.101 /24
#--- Set the default gateway for the in-path interfaces to be the LAN
#--- side Layer-3 switch HSRP VIP
ip in-path-gateway inpath0_0 "10.0.0.254"
ip in-path-gateway inpath0_1 "10.0.0.254"
#--- Enable Enhanced auto-discovery, enabled by default on new factory installs of
#--- RiOS v6.x. If you have upgraded a Steelhead appliance to that level or more recently, you
#--- will need to apply this command
in-path peering auto
#--- Simplified Routing destination only should be used and
#--- is on by default with new RiOS v6.x installs
in-path simplified routing dest-only
```

Configuring Multiple WAN Router Deployments with Connection Forwarding

The following section describes best practices for parallel Steelhead appliance deployments at locations with multiple routers. Each of the scenarios that follow can be modified to use additional Steelhead appliances for each path to the WAN, using either the master and backup or serial cluster configurations. If you are using multiple Steelhead appliances on each path, every Steelhead appliance at the location must be configured as a connection-forwarding neighbor for every other Steelhead appliance at the location. This section covers the following scenarios:

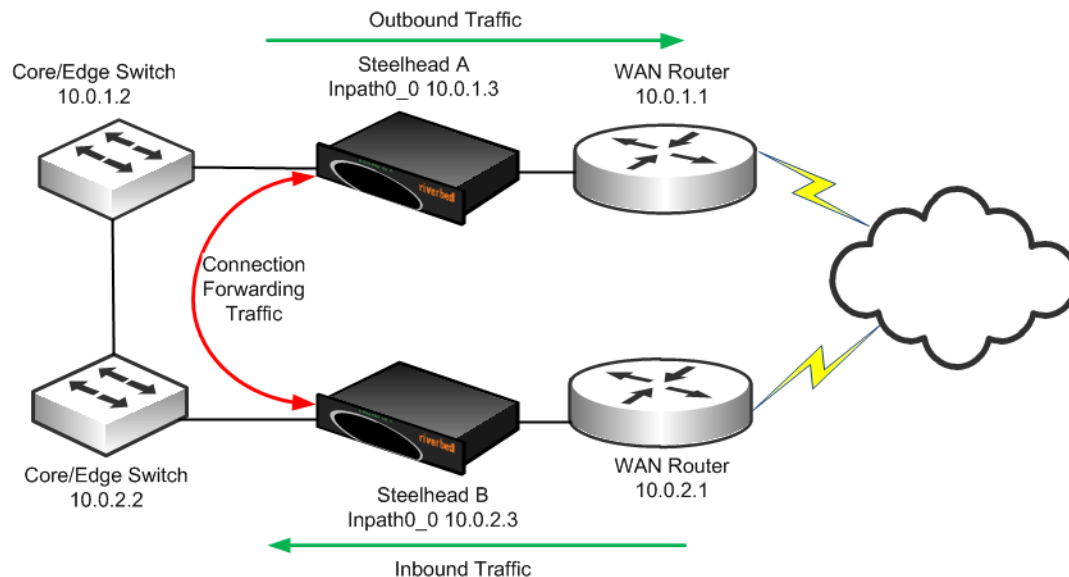
- [“Configuring Basic Connection Forwarding” on page 199](#)
- [“Configuring Connection Forwarding with Allow-Failure and Fail-to-Block” on page 200](#)

- “Configuring a Dual Steelhead Appliance and Dual Layer-2 Switches Deployment” on page 201
- “Configuring a Dual Steelhead Appliance and Dual Layer-3 Switches Deployment” on page 203
- “Configuring a Dual Steelhead Appliance with Multiple In-Path Deployment” on page 204

Configuring Basic Connection Forwarding

This example requires you to have configured your cabling and duplex according to the recommendations described in “Cabling and Duplex” on page 177.

Figure 7-14. Physical In-Path Deployment with Connection Forwarding



This example makes the following assumptions:

- Connection forwarding is enabled by configuring the in-path0_0 IP address of the two Steelhead appliances as neighbors.
- When one of the Steelhead appliances fails, the neighbor Steelhead appliance stops attempting to optimize new connections until the down Steelhead appliance recovers or is replaced.
- Simplified routing removes any packet ricochet that might occur when the Steelhead appliance sends traffic to remote Steelhead appliances.

To configure connection-forwarding multiple WAN routers

1. On Steelhead A, connect to the CLI and enter the following commands:

```
#--- Enable and configure the in-path Interfaces
in-path enable
interface inpath0_0 ip address 10.0.1.3 /24
#--- Set the default gateway for the in-path interface to be the LAN
#--- side Layer-3 switch
ip in-path-gateway inpath0_0 10.0.1.2
#--- Enable Enhanced auto-discovery, enabled by default on new factory installs of
#--- RiOS v6.x. If you have upgraded a Steelhead appliance to that level or more recently, you
#--- will need to apply this command
in-path peering auto
#--- Simplified Routing destination only should be used and
#--- is on by default with new RiOS v6.x installs
in-path simplified routing dest-only
```

```
#--- Enables Connection Forwarding to neighbor 10.0.2.3
steelhead communication enable
steelhead name SteelheadB main-ip 10.0.2.3
```

2. On Steelhead B, connect to the CLI and enter the following commands:

```
#--- Enable and configure the in-path Interfaces
in-path enable
interface inpath0_0 ip address 10.0.2.3 /24
#--- Set the default gateway for the in-path interface to be the LAN
#--- side Layer-3 switch
ip in-path-gateway inpath0_0 10.0.2.2
#--- Enable Enhanced auto-discovery, enabled by default on new factory installs of
#--- RiOS v6.x. If you have upgraded a Steelhead appliance to that level or more recently, you
#--- will need to apply this command
in-path peering auto
#--- Simplified Routing destination only should be used and
#--- is on by default with new RiOS v6.x installs
in-path simplified routing dest-only
#--- Enables Connection Forwarding to neighbor 10.0.2.3
steelhead communication enable
steelhead name SteelheadA main-ip 10.0.1.3
```

Note: These steps do not include the configuration of features such as duplex, alarms, and DNS.

For details on connection forwarding, see [“Connection Forwarding” on page 42](#).

Configuring Connection Forwarding with Allow-Failure and Fail-to-Block

This example requires you to have configured your cabling and duplex according to the recommendations described in [“Cabling and Duplex” on page 177](#).

The following example represents the minimum steps required to configure a Steelhead appliance deployment in which connection forwarding is configured, and both **fail-to-block** and **allow-failure** are enabled. This example does not include configuration instruction for features such as the management interface, DNS, and SNMP.

This example makes the following assumptions:

- Connection forwarding is enabled by configuring the in-path0_0 IP address of the two Steelhead appliances as neighbors.
- Fail-to-block option is enabled. (This is not supported with all in-path hardware and Steelhead appliance models.)
- The **allow-failure** CLI command is enabled. This specifies that a Steelhead B continues to optimize new connections, if Steelhead appliance A down.
- Simplified routing removes any packet ricochet that might occur when the Steelhead appliance sends traffic to remote Steelhead appliances.

To configure connection forwarding with multiple WAN routers, allow-failure, and fail-to-block

1. On Steelhead A, connect to the CLI and enter the following commands:

```
#--- Enable and configure the in-path Interfaces
in-path enable
interface inpath0_0 ip address 10.0.1.3 /24
#--- Set the default gateway for the in-path interface to be the LAN
#--- side Layer-3 switch
```

```

ip in-path-gateway inpath0_0 10.0.1.2
#--- Enable Enhanced auto-discovery, enabled by default on new factory installs of
#--- RiOS v6.x. If you have upgraded a Steelhead appliance to that level or more recently, you
#--- will need to apply this command
in-path peering auto
#--- Simplified Routing destination only should be used and
#--- is on by default with new RiOS v6.x installs
in-path simplified routing dest-only
#--- Enable Connection Forwarding to neighbor 10.0.2.3
#--- allow-failure allows the Steelhead appliance to continue optimizing
#--- traffic even if the neighbor is down
steelhead communication enable
steelhead name SteelheadB main-ip 10.0.2.3
steelhead communication allow-failure
#--- Enable fail-to-block on inpath0_0
no interface inpath0_0 fail-to-bypass enable

```

2. On Steelhead B, connect to the CLI and enter the following commands:

```

#--- Enable and configure the in-path Interfaces
in-path enable
interface inpath0_0 ip address 10.0.2.3 /24
#--- Set the default gateway for the in-path interface to be the LAN
#--- side Layer-3 switch
ip in-path-gateway inpath0_0 10.0.2.2
#--- Enable Enhanced auto-discovery, enabled by default on new factory installs of
#--- RiOS v6.x. If you have upgraded a Steelhead appliance to that level or more recently, you
#--- will need to apply this command
in-path peering auto
#--- Simplified Routing destination only should be used and
#--- is on by default with new RiOS v6.x installs
in-path simplified routing dest-only
#--- Enable Connection Forwarding to neighbor 10.0.1.3
#--- allow-failure allows the Steelhead appliance to continue optimizing
#--- traffic even if the neighbor is down
steelhead communication enable
steelhead name SteelheadA main-ip 10.0.1.3
steelhead communication allow-failure
#--- Enable fail-to-block on inpath0_0
no interface inpath0_0 fail-to-bypass enable

```

Note: These steps do not include the configuration of features such as duplex, alarms, and DNS.

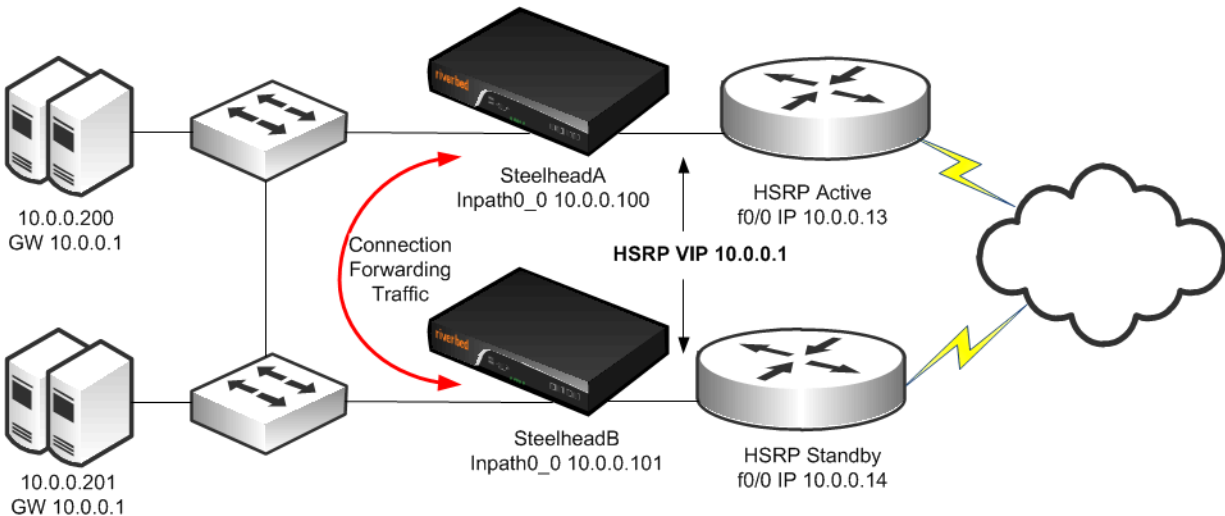
For details on connection forwarding, see [“Connection Forwarding”](#) on page 42.

Configuring a Dual Steelhead Appliance and Dual Layer-2 Switches Deployment

[Figure 7-15](#) shows a topology in which there are two routers, two Layer-2 switches, and two Steelhead appliances at the remote location. The client and the Steelhead appliances are all in the same subnet. The client uses the HSRP virtual IP as its default gateway (10.0.0.1).

In this environment, the in-path gateway on both Steelhead appliances must point to the HSRP virtual IP (10.0.0.1). You do not need to enable simplified routing because the client is on the same subnet as the Steelhead appliance. You must configure connection forwarding between the two Steelhead appliances. The connection forwarding path must use the LAN interface of the Steelhead appliances.

Figure 7-15. Dual Steelhead Appliances, Dual Layer-2 Switches, Dual Router Deployment



To configure dual Steelhead appliances, dual Layer-2 switches, and dual routers

1. On Steelhead A, connect to the CLI and enter the following commands:

```
#--- Enable and configure the in-path Interfaces
in-path enable
interface inpath0_0 ip address 10.0.0.100 /24
#--- Set the default gateway for the in-path interfaces to be the WAN
#--- side HSRP VIP
ip in-path-gateway inpath0_0 10.0.0.1
#--- Enable Enhanced auto-discovery, enabled by default on new factory installs of
#--- RiOS v6.x. If you have upgraded a Steelhead appliance to that level or more recently, you
#--- will need to apply this command
in-path peering auto
#--- Simplified Routing is not required but can be enabled. (Simplified Routing
#--- destination only is on by default with new RiOS v6.x installs)
in-path simplified routing dest-only
#--- Enable Connection Forwarding to neighbor 10.0.0.101
#--- allow-failure allows the Steelhead appliance to continue optimizing
#--- traffic even if the neighbor is down
steelhead communication enable
steelhead name SteelheadB main-ip 10.0.0.101
steelhead communication allow-failure
#--- Enable fail-to-block on inpath0_0
no interface inpath0_0 fail-to-bypass enable
```

2. On Steelhead B, connect to the CLI and enter the following commands:

```
#--- Enable and configure the in-path Interfaces
in-path enable
interface inpath0_0 ip address 10.0.0.101 /24
#--- Set the default gateway for the in-path interfaces to be the WAN
#--- side HSRP VIP
ip in-path-gateway inpath0_0 10.0.0.1
#--- Enable Enhanced auto-discovery, enabled by default on new factory installs of
#--- RiOS v6.x. If you have upgraded a Steelhead appliance to that level or more recently, you
```

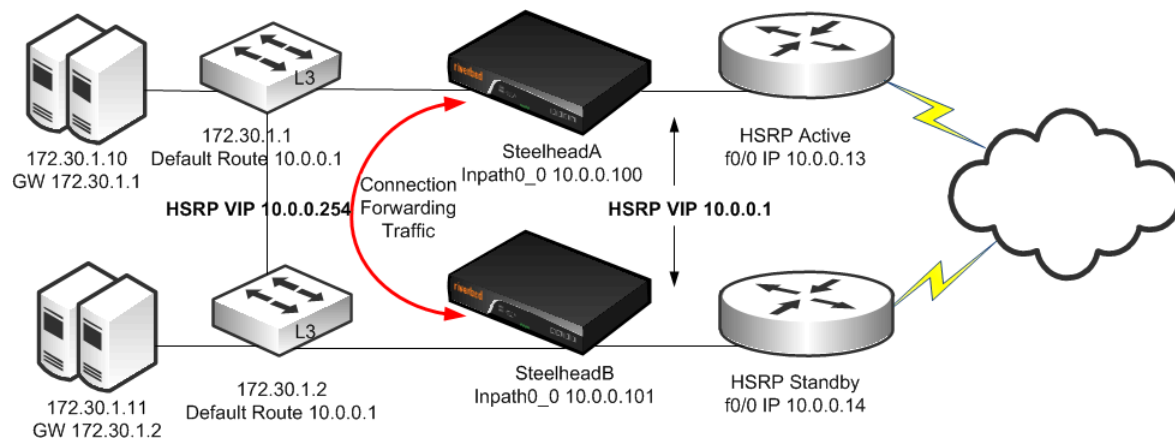
```
#--- will need to apply this command
in-path peering auto
#--- Simplified Routing is not required but can be enabled. (Simplified Routing
#--- destination only is on by default with new RiOS v6.x installs)
in-path simplified routing dest-only
#--- Enable Connection Forwarding to neighbor 10.0.0.100
#--- allow-failure allows the Steelhead appliance to continue optimizing
#--- traffic even if the neighbor is down
steelhead communication enable
steelhead name SteelheadA main-ip 10.0.0.100
steelhead communication allow-failure
#--- Enable fail-to-block on inpath0_0
no interface inpath0_0 fail-to-bypass enable
```

Configuring a Dual Steelhead Appliance and Dual Layer-3 Switches Deployment

Figure 7-16 shows a topology in which there are two routers, two Layer-3 switches, and two Steelhead appliances at the remote location. The clients and the Steelhead appliances are in different subnets. The clients use the Layer-3 switch as the default gateway. The Layer-3 switch does not have any routing protocols configured and relies on the default route to reach other subnets. The default route uses the HSRP IP address as the next hop.

In this environment, the in-path gateway on both Steelhead appliances must use the HSRP address of the Layer-3 switches as its default gateway (10.0.0.254) while configuring simplified routing to populate its table based on destination MAC address (CLI command: **in-path simplified routing dest-only**). You must configure connection forwarding between the two Steelhead appliances. The connection forwarding path must use the LAN interface of the Steelhead appliances.

Figure 7-16. Dual Steelhead Appliances, Dual Layer-3 Switches, Static Routing, Dual Router Deployment



To configure dual Steelhead appliances, dual Layer-3 switches, static routing, and dual routers

1. On Steelhead A, connect to the CLI and enter the following commands:

```
#--- Enable and configure the in-path Interfaces
in-path enable
interface inpath0_0 ip address 10.0.0.100 /24
#--- Set the default gateway for the in-path interfaces to be the LAN
#--- side Layer-3 switch HSRP VIP
ip in-path-gateway inpath0_0 10.0.0.254
#--- Enable Enhanced auto-discovery, enabled by default on new factory installs of
#--- RiOS v6.x. If you have upgraded a Steelhead appliance to that level or more recently, you
#--- will need to apply this command
in-path peering auto
#--- Simplified Routing destination only should be used and
```

```
#--- is on by default with new RiOS v6.x installs
in-path simplified routing dest-only
#--- Enable Connection Forwarding to neighbor 10.0.0.101
#--- allow-failure allows the Steelhead appliance to continue optimizing
#--- traffic even if the neighbor is down
steelhead communication enable
steelhead name SteelheadB main-ip 10.0.0.101
steelhead communication allow-failure
#--- Enable fail-to-block on inpath0_0
no interface inpath0_0 fail-to-bypass enable
```

2. On Steelhead B, connect to the CLI and enter the following commands:

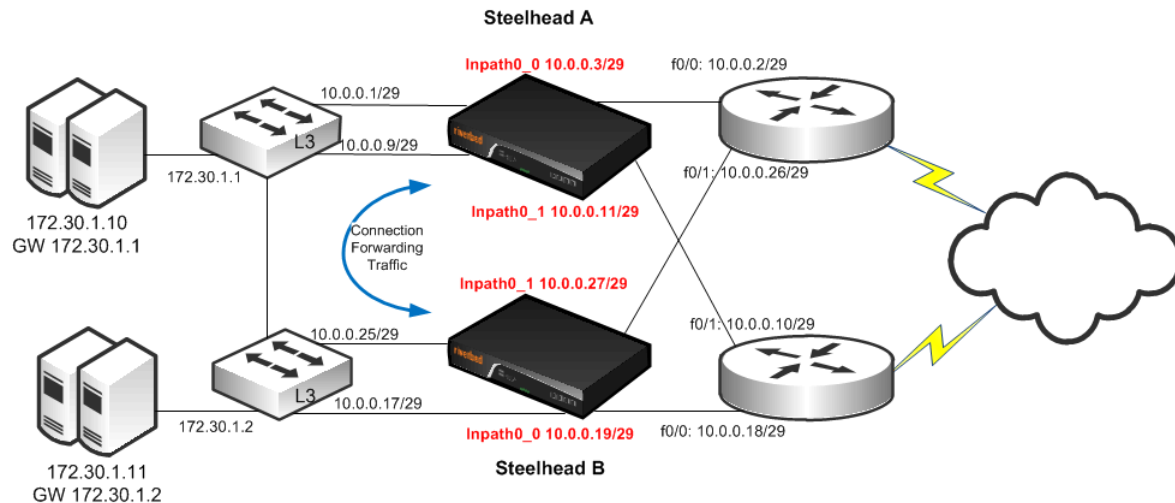
```
#--- Enable and configure the in-path Interfaces
in-path enable
interface inpath0_0 ip address 10.0.0.101 /24
#--- Set the default gateway for the in-path interfaces to be the LAN
#--- side Layer-3 switch HSRP VIP
ip in-path-gateway inpath0_0 10.0.0.254
#--- Enable Enhanced auto-discovery, enabled by default on new factory installs of
#--- RiOS v6.x. If you have upgraded a Steelhead appliance to that level or more recently, you
#--- will need to apply this command
in-path peering auto
#--- Simplified Routing destination only should be used and
#--- is on by default with new RiOS v6.x installs
in-path simplified routing dest-only
#--- Enable Connection Forwarding to neighbor 10.0.0.100
#--- allow-failure allows the Steelhead appliance to continue optimizing
#--- traffic even if the neighbor is down
steelhead communication enable
steelhead name SteelheadA main-ip 10.0.0.100
steelhead communication allow-failure
#--- Enable fail-to-block on inpath0_0
no interface inpath0_0 fail-to-bypass enable
```

Configuring a Dual Steelhead Appliance with Multiple In-Path Deployment

Figure 7-17 shows a topology similar to Figure 7-16, but each router has two links going back to the Layer-3 switches totaling four links. Each link between the router and switch are independent /29 bit networks. You can configure dynamic routing protocols so that traffic can flow inbound or outbound. You can use this network design to prevent the loss of a single Layer-3 switch from cutting connectivity to the attached router. To ensure traffic is routed to the partner Steelhead appliance during a failure, you can use the two in-path interfaces and deploy the Steelhead appliances in parallel, (requires larger model than 150, 250 or 550), with connection forwarding, and fail-to-block.

The clients and the Steelhead appliances are in different subnets and simplified routing is enabled (**in-path simplified routing dest-only**). To ensure connection forwarding traffic is sent to the LAN side, Riverbed recommends that you configure each in-path interface default gateway to point to the LAN side, the Layer-3 switch.

Figure 7-17. Dual Steelhead Appliances with Multiple In-Path Deployment



To configure dual Steelhead appliances with multiple in-path interfaces

1. On Steelhead A, connect to the CLI and enter the following commands:

```
#--- Enable and configure the in-path interfaces
in-path enable
in-path interface inpath0_0 enable
interface inpath0_0 ip address 10.0.0.3 /29
in-path interface inpath0_1 enable
interface inpath0_1 ip address 10.0.0.11 /29
#--- Set the default gateway for the in-path interfaces to be the LAN side Layer-3 Switch
ip in-path-gateway inpath0_0 10.0.0.1
ip in-path gateway inpath0_1 10.0.0.9
#--- Enable enhanced auto-discovery, enabled by default on new factory installs of
#--- RiOS v6.x. If you have upgraded a Steelhead appliance to that level or more recently, you
#--- will need to apply this command
in-path peering auto
#--- simplified routing destination only should be used and
#--- is on by default with new RiOS v6.x installs
in-path simplified routing dest-only
#--- Enable Connection Forwarding to Steelhead appliance neighbor, specifying both neighbors
#--- in-paths
#--- Enable multi-interface support
#--- Allow-failure allows the Steelhead appliance to continue optimizing traffic even
#--- if neighbor is down
steelhead communication enable
steelhead communication multi-interface enable
steelhead name SteelheadB main-ip 10.0.0.19
steelhead name SteelheadB additional-ip 10.0.0.27
steelhead communication allow-failure
#--- Enable fail-to-block on inpath0_0 and inpath0_1
no interface inpath0_0 fail-to-bypass enable
no interface inpath0_1 fail-to-bypass enable
```

2. On Steelhead B, connect to the CLI and enter the following commands:

```
#--- Enable and configure the in-path Interfaces
```



```

in-path enable
in-path interface inpath0_0 enable
interface inpath0_0 ip address 10.0.0.19 /29
in-path interface inpath0_1 enable
interface inpath0_1 ip address 10.0.0.27 /29
#--- Set the default gateway for the in-path interfaces to be the LANside L3 Switch
ip in-path-gateway inpath0_0 10.0.0.17
ip in-path gateway inpath0_1 10.0.0.25
#--- Enable Enhanced auto-discovery, enabled by default on new factory installs of
#--- RiOS v6.x. If you have upgraded a Steelhead appliance to that level or more recently, you
#--- will need to apply this command
in-path peering auto
#--- Simplified Routing destination only should be used and
#--- is on by default with new RiOS v6.x installs
in-path simplified routing dest-only
#--- Enable Connection Forwarding to Steelhead appliance neighbor, specifying both neighbors
#--- in-paths
#--- Enable multi-interface support
#--- Allow-failure allows the Steelhead appliance to continue optimizing traffic even if
#--- neighbor is down
steelhead communication enable
steelhead communication multi-interface enable
steelhead name SteelheadA main-ip 10.0.0.3
steelhead name SteelheadA additional-ip 10.0.0.11
steelhead communication allow-failure
#--- Enable fail-to-block on inpath0_0 and inpath0_1
no interface inpath0_0 fail-to-bypass enable
no interface inpath0_1 fail-to-bypass enable

```

You can configure and deploy routers in an endless number of variations, for example:

- The links between Layer-3 switches and routers can be Layer-2 instead of separate Layer-3 networks.
- Layer-3 switches that are WAN distribution switches connected to a pair of core switches, then connected to other distribution switches before connecting to the end hosts.
- Instead of two routers, you can use four routers connecting to various WAN providers for a total of eight connections to the Layer-3 switches.
- You can use static routes, or any number of routing protocols, to balance traffic across multiple paths.

Regardless of the variations, the same deployment logic applies. The Steelhead appliances must cover all the desired paths of traffic to be optimized. You can use up to 10 supported in-path interfaces on the larger Steelhead appliances. If you have parallel Steelhead appliances, follow the same simplified routing and connection forwarding guidelines in this section. For information about serial clusters, see [“Configuring Serial Cluster Deployments” on page 187](#).

802.1Q Trunk Deployments

This section describes the use of virtual LANs (VLANs) and 802.1Q, which allows multiple logical networks to span a single physical link. IEEE 802.1Q is a networking standard that allows multiple bridged networks to transparently share the same physical network. IEEE 802.1Q is also referred to as *VLAN Tagging*, and *dot1q*.

This section includes the following topics:

- [“Overview of VLAN Trunk” on page 207](#)
- [“Configuring a Steelhead Appliance on an 802.1Q Trunk Link” on page 208](#)
- [“Capturing Network Traces Using tcpdump” on page 209](#)

The Steelhead appliance does not support overlapping IP address spaces, even if the overlapping IPs are kept separate through VLAN tags.

For details on alternative configurations, see [“VPN Routing and Forwarding” on page 323](#).

Overview of VLAN Trunk

A Steelhead appliance can be deployed physically in-path on an 802.1Q trunk link, and can optimize connections where packets have been tagged with an 802.1Q header. As in other physical in-path deployments, the Steelhead appliances in-path interface must be configured with an IP address and a default gateway. If the Steelhead appliances in-path IP address is in a subnet whose traffic is normally tagged when present on the in-path link, the Steelhead appliance's in-path interface must be configured with the VLAN for that subnet. This allows the Steelhead appliance to appropriately tag packets transmitted from the in-path interface that use the in-path IP address as the source address. The Steelhead appliance can optimize traffic on the VLANs different from the VLAN containing the in-path IP address.

Steelhead appliances can be deployed across multiple 802.1Q trunk links. Each in-path interface requires:

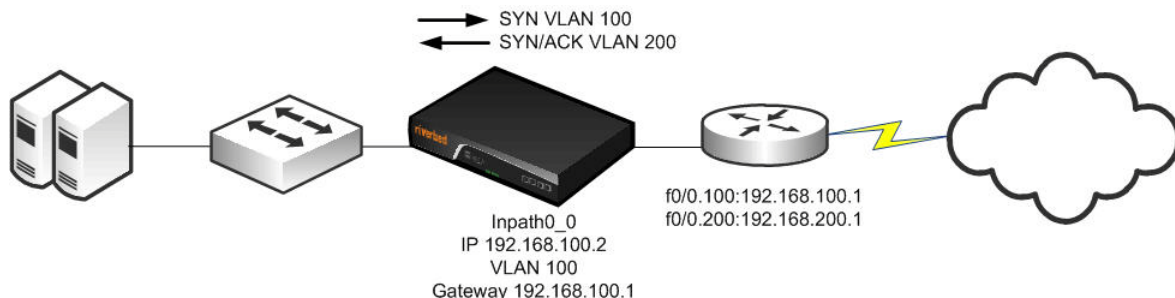
- an IP address.
- a default gateway.
- a VLAN ID (if required for the in-path IP address subnet).

Steelhead appliances configured as connection-forwarding neighbors can be deployed on 802.1Q trunk links.

Steelhead appliances maintain the VLAN ID when transmitting packets for the LAN-side of optimized connections. If correct addressing or port transparency is used, the Steelhead appliance uses the configured in-path VLAN ID when transmitting packets towards the WAN. When using the full address transparency WAN visibility mode (for details, see [“Full Address Transparency” on page 53](#)), Steelhead appliances maintain the VLAN ID (along with IP address and TCP ports) when transmitting packets on the WAN-side of optimized connections. The Steelhead appliance can maintain the VLAN IDs even if there is a difference between the packets going to the WAN and those returning from the WAN. You do not have to configure the VLAN ID on the in-path interface to match any of those seen for optimized connections.

For example, assume that Steelhead appliance is configured for full address transparency. The SYN packet for a TCP connection traveling toward the WAN has a VLAN ID of 100, and the SYN/ACK packet returning from the WAN has a VLAN of 200. If a Steelhead appliance optimizes this connection, then the Steelhead appliance transmits packets toward the LAN using VLAN ID 200. The Steelhead appliance monitors the MAC addresses associated with each VLAN.

Figure 7-18. Example of Basic VLAN Trunk Deployment



Maintaining the VLAN IDs in this manner requires using the **vlan-conn-based** and **mac-match-vlan** CLI commands, which are enabled by default in RiOS v6.0 or later, and the other commands listed in the following configuration example.

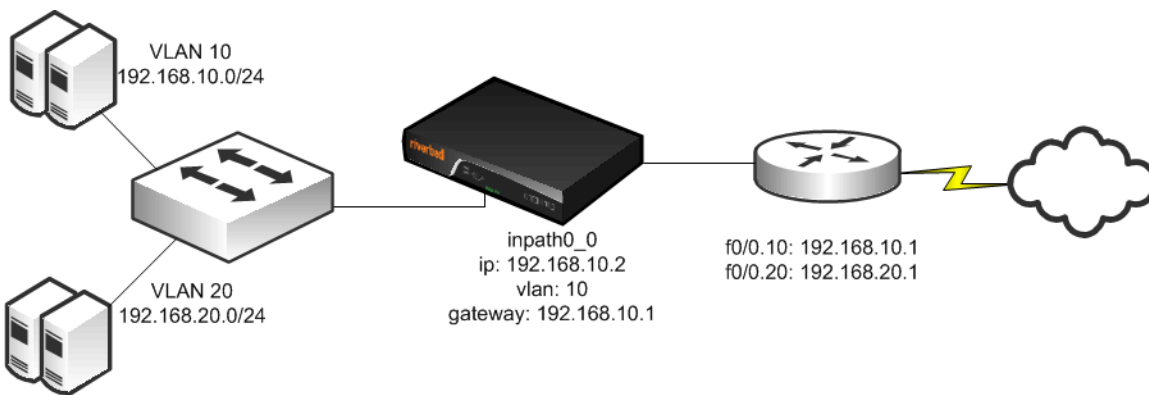
For details, see [“Configuring a Steelhead Appliance on an 802.1Q Trunk Link”](#) on page 208.

Both the in-path and peering rules can use VLAN tags as matching parameters. This allows an administrator to control optimization based on VLAN tags, along with other information such as IP addresses, subnets, or TCP ports.

Configuring a Steelhead Appliance on an 802.1Q Trunk Link

Figure 7-19 shows a Steelhead appliance deployed physically in-path on an 802.1Q trunk link. Two VLANs are present on the in-path link: VLAN 10, which contains subnet 192.168.10.0/24, and VLAN 20, which contains subnet 192.168.20.0/24. The Steelhead appliance is given an in-path IP address in the 192.168.10.0/24 subnet. The Steelhead appliance is configured to use VLAN 10 for its in-path interface, and to use the router subinterface IP address as its default gateway. Even though the Steelhead appliance has an in-path IP address in subnet 192.168.10.0/24 and VLAN 10, it can optimize traffic in VLAN 20.

Figure 7-19. Steelhead Appliance Deployed Physically In-Path on an 802.1Q Trunk Link



To deploy a Steelhead appliance physically in-path on an 802.1Q trunk link

1. On the Steelhead appliance, connect to the CLI and enter the following commands:

```
#--- Enable and configure the in-path Interface
in-path enable
interface inpath0_0 ip address 192.168.10.2 /24
#--- Set the default gateway for the inpath0_0 interface to be the WAN
#--- side router VLAN 10 interface
ip in-path-gateway inpath0_0 192.168.10.1
#--- Assign VLAN 10 to the inpath0_0 interface
in-path interface inpath0_0 vlan 10
#--- Enable Simplified Routing All. (Simplified Routing destination only is on
#--- by default with new RiOS v6.x installs)
in-path simplified routing all
#--- New factory installs of RiOS v6.0.1 and greater will have all of the following
#--- configuration parameters set already. If you have upgraded a Steelhead appliance to that
#--- level or more recent, you will need to apply some or all of the following.
#--- Enable Enhanced auto-discovery
in-path peering auto
#--- Ensure the Steelhead appliance performs autodiscovery probing for the
#--- FTP and MAPI data channels.
in-path probe-ftp-data
in-path probe-mapi-data
#--- Allow static routes to take precedence over Simplified Routing
in-path simplified mac-def-gw-only
#--- Ensure Steelhead appliance transmits to LAN hosts with the same vlan tags as received
in-path mac-match-vlan
in-path vlan-conn-based
```

```
#--- Ensure autodiscovery probing happens whenever possible, to learn vlan to IP mappings.
no in-path peer-probe-cach
no in-path probe-caching enable
#--- Ensure interoperability with remote Steelhead appliances whose in-path address is
#--- in the same subnet or vlan
in-path mac-except-locl
```

Capturing Network Traces Using tcpdump

Use caution when using network traces on trunk links. If you configure a TCP dump filter that restricts the captured packets to a specified set (for example, based on IP addresses or ports), by default tcpdump does not capture packets with an 802.1Q tag.

To capture packets with an 802.1Q tag, you must prefix the filter string with the keyword **vlan**. Enter the following CLI command:

```
tcpdump -i wanX_Y vlan and host
```

For details, see the *Riverbed Command-Line Interface Reference Manual*.

Layer-2 WAN Deployments

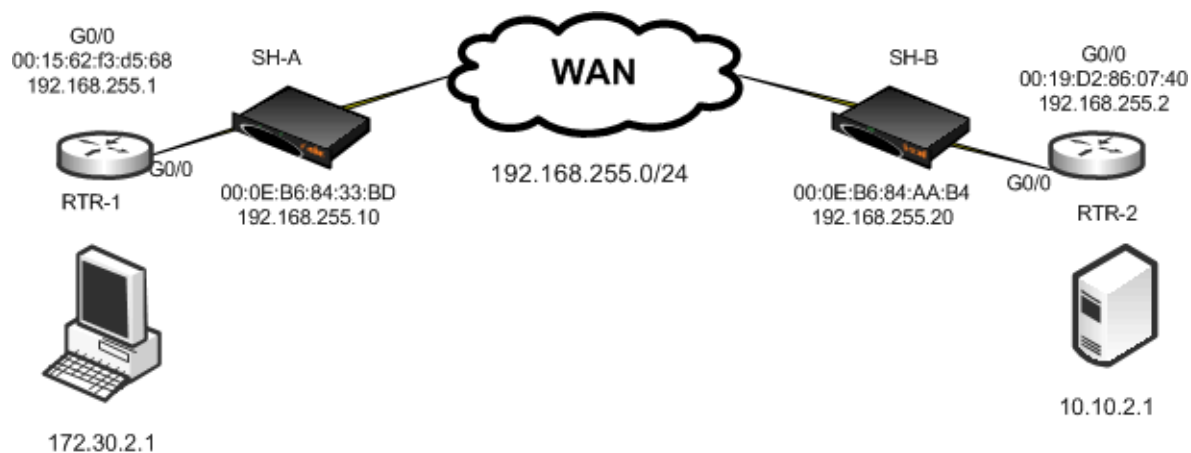
The types of Layer-2 WANs are as follows:

- “Layer-2 WANs” on page 209
- “Broadcast Layer-2 WANs” on page 210

Layer-2 WANs

On a Layer-2 WAN, Riverbed recommends setting the LAN router closest to the Steelhead appliance as its in-path default gateway. For example, in [Figure 7-20](#), Steelhead appliance A must use 192.168.255.1 as its in-path default gateway and Steelhead appliance B must use 192.168.255.2 as its in-path default gateway.

Figure 7-20. Steelhead Appliances Deployed in a Layer-2 WAN



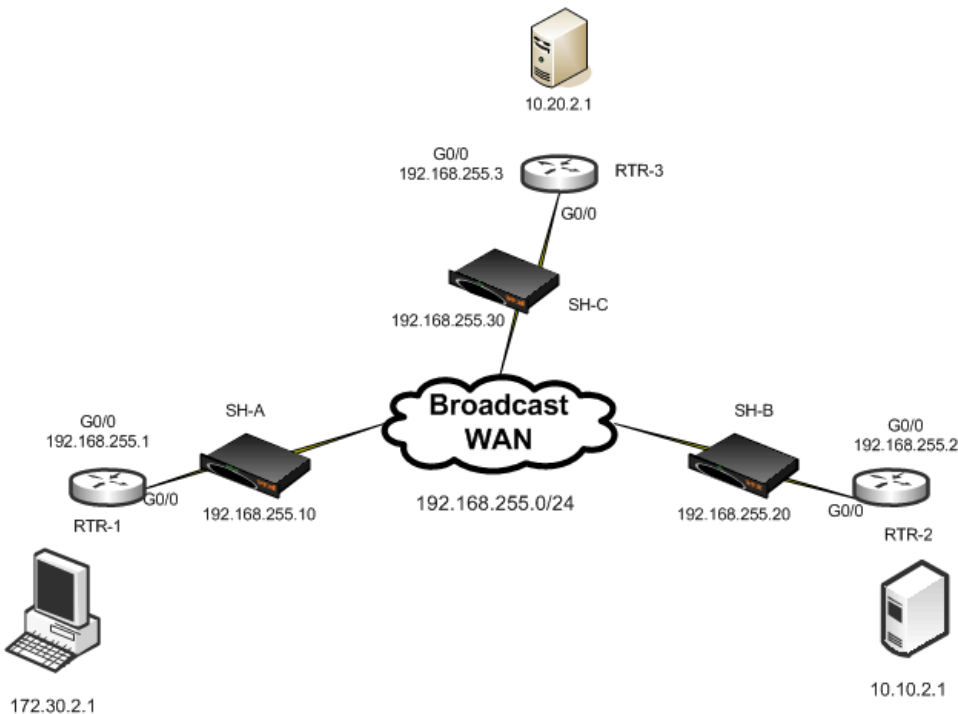
If the links to the WAN carry 802.1Q tagged packets, and if the WAN service provider routes packets based on 802.1Q tags, it is necessary to configure the Steelhead appliances for 802.1Q deployment, and to use the full address transparency mode.

For details on full address transparency mode, see [“Full Address Transparency”](#) on page 53.

Broadcast Layer-2 WANs

A Layer-2 broadcast network over the WAN is very similar to a Layer-2 WAN. However, a Layer-2 broadcast WAN behaves like a hub, whereby packets from each site are replicated over the WAN to all the other sites. For example, in [Figure 7-21](#), Router 2 can detect traffic between 172.30.2.1 and 10.10.2.1.

Figure 7-21. A Layer-2 Broadcast WAN Deployment



When deploying Steelhead appliances in a Layer-2 broadcast WAN, you must ensure that the correct Steelhead appliance is handling the traffic. Furthermore, a Layer-2 Broadcast WAN is not compatible with simplified routing and therefore you must use static routes to avoid packet ricochet.

In [Figure 7-21](#), when 172.30.2.1 sends a SYN packet toward 10.10.2.1, Steelhead appliance A adds its probe option to the SYN packet (SYN+). The correct Steelhead appliance that responds to this packet is Steelhead appliance B, because it is closest to the 10.10.2.1 server. However, because of the Layer-2 Broadcast WAN, Steelhead appliance C would also detect the SYN+ packet and respond to Steelhead appliance A. This behavior creates two probe responses (SYN/ACK+) from two separate Steelhead appliances.

If the latency between Steelhead appliance A and Steelhead appliance C is lower than that of Steelhead appliance A and Steelhead appliance B, then Steelhead appliance A receives the probe response from Steelhead appliance C first. When the probe response Steelhead appliance B arrives at Steelhead appliance A, Steelhead appliance A ignores the probe response, because it has already received a probe response from Steelhead appliance C.

This is clearly an undesirable situation, because Steelhead appliance A must be peering with Steelhead appliance B and not Steelhead appliance C when trying to reach the 10.10.2.1 server. To avoid this situation, enter the following CLI command:

```
in-path broadcast support enable
```

When broadcast support is enabled, the Steelhead appliance checks its routing table to see whether it uses its LAN or WAN interface to reach the destination IP. If the destination IP is reachable through the LAN, the appliance sends a probe response to the sender. Otherwise, it simply ignores the probe request.

Alternatively, you can use a fixed-target rule to define the Steelhead appliance peers. However, fixed-target rules might not be scalable for larger deployments.

VLAN Bridging Deployments

This section describes the use of virtual LAN (VLAN) bridging. This section includes the following topics:

- [“Overview of VLAN Bridging Deployment” on page 211](#)
- [“VLAN Bridging Considerations” on page 212](#)
- [“VLAN Bridging Variations” on page 213](#)

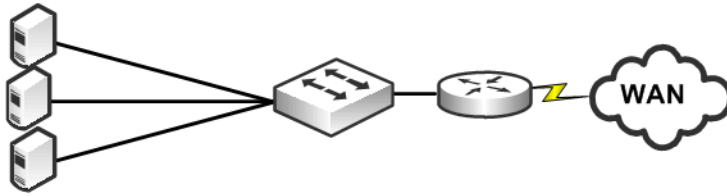
Overview of VLAN Bridging Deployment

The term *VLAN bridging* refers to a network design in which both the LAN and WAN ports of a Steelhead appliance's in-path interface are connected to a single switch or router. The switch or router is then configured so that traffic to be optimized must pass through the Steelhead appliance—by forcing the traffic's Layer-2 path to or from the WAN to pass through the in-path interface.

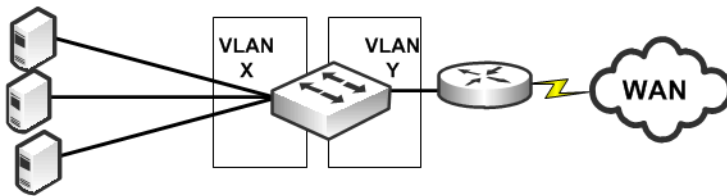
VLAN bridging is useful in network environments in which it is difficult to install a Steelhead appliance physically in-path. For example, if fiber interfaces are needed for a physical in-path installation, but only copper interfaces are available on the Steelhead appliance, you can use VLAN bridging as a simpler alternative to WCCP or PBR.

Figure 7-22 shows the principles of VLAN bridging. An existing switch or router is divided into two separate VLANs, and the Steelhead appliance's LAN and WAN interfaces are used as the Layer-2 bridge that connects the VLANs.

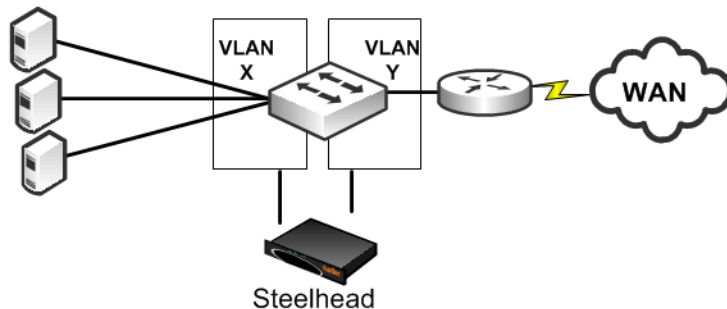
Figure 7-22. VLAN Bridging Principles



Switch divided into 2 VLANs



VLANs Bridged



VLAN Bridging Considerations

A VLAN bridging deployment has the same features and functionality as a physically in-path Steelhead appliance, with the exception of 802.1Q VLAN trunking.

Consider the following when you use a VLAN bridging deployment:

- You can use an 802.1Q trunk with VLAN bridging between multiple VLANs on the same in-path interface, but this requires switch-specific features.

For details on multiple VLANs on the same in-path interface, see [“Multiple VLAN Bridging with VLAN Mapping” on page 214](#).

- Use the same cables for the WAN and LAN interfaces—the same as you use for physical in-path deployments.

For details on cables, see [“Cabling and Duplex” on page 177](#).

- The switch detects the same MAC addresses in two different VLANs. Because most switches have separate MAC address tables per VLAN (independent VLAN learning, or IVL), some older switches can have only one MAC table for all VLANs (shared VLAN learning). Use only switches that have IVL with VLAN bridging.

- Verify that the switch allows access to its management IP address from multiple VLANs. Avoid using a switch whose management IP is only reachable from the default VLAN, because this prevents managing the switch. Some switches assign their management IP address to the default VLAN, and cannot be altered—for example, the Cisco 2950 switch.

VLAN Bridging Variations

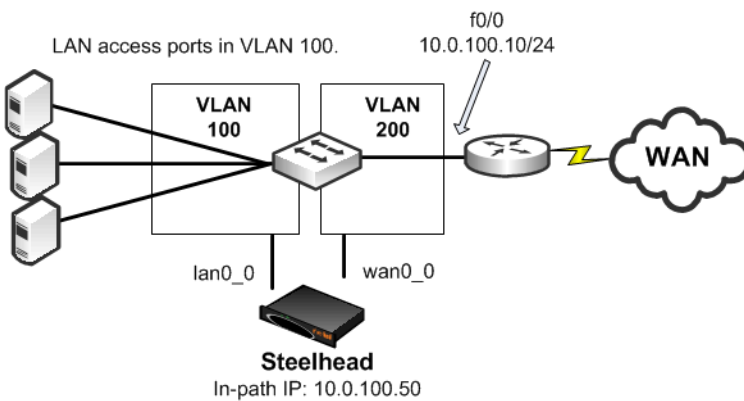
The variations of VLAN-Bridging are as follows:

- [“Layer-2 VLAN Bridging” on page 213](#)
- [“Layer-3 VLAN Bridging” on page 214](#)
- [“Multiple VLAN Bridging with VLAN Mapping” on page 214](#)

Layer-2 VLAN Bridging

In a Layer-2 VLAN bridging deployment, the Steelhead appliance is connected by VLANs on the Layer-2 switch. All traffic is bridged through the Steelhead appliances as it passes to and from the WAN routers. [Figure 7-23](#) shows a Layer-2 VLAN bridging deployment.

Figure 7-23. Layer-2 VLAN Bridging



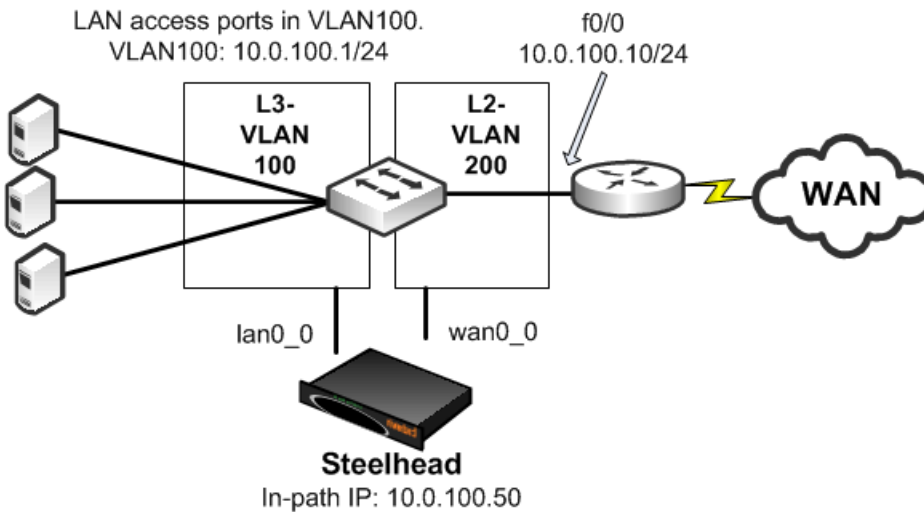
Note the following:

- VLAN 100 and VLAN 200 are Layer-2 VLANs.
- The default gateway of the hosts on the LAN must point to the router interface IP address.
- VLAN 100 contains the switch ports of the hosts and the switch port connected to the lan0_0 interface of the Steelhead appliance.
- VLAN 200 contains the switch ports, the router, and the wan0_0 interface of the Steelhead appliance.
- The default gateway of the Steelhead appliance is the IP address of the WAN router.

Layer-3 VLAN Bridging

In a Layer-3 VLAN bridging deployment, the Steelhead appliance is connected across Layer-3 and Layer-2 VLANs on a Layer-2/Layer-3 switch. All traffic is switched through the Steelhead appliance as it passes to and from the WAN router. [Figure 7-24](#) shows a Layer-3 VLAN bridging deployment.

Figure 7-24. Layer-3 VLAN Bridging



Note the following:

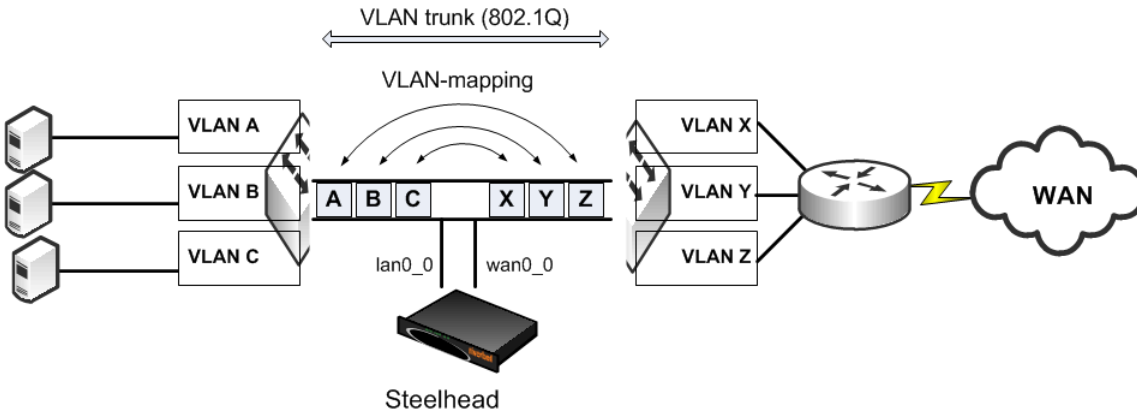
- Hosts on the VLAN 100 must point to VLAN 100 IP address as the default gateway.
- VLAN 100 contains the switch ports of the hosts, and the switch port connected to the lan0_0 interface of the Steelhead appliance.
- VLAN 200 contains the switch ports, the router, and the wan0_0 interface of the Steelhead appliance connect to.
- The default gateway of the Steelhead appliance is the IP address of the WAN router.

Multiple VLAN Bridging with VLAN Mapping

A limitation in Layer-2 VLAN bridging and Layer-3 VLAN bridging is that a single in-path interface can only bridge two different VLANs. If the Steelhead appliance-connected switch ports are configured to be 802.1Q trunks (so that many VLANs can be sent or received), the switch does not bridge traffic for the VLANs across the ports.

To connect to multiple VLANs, you need a switch that supports *VLAN mapping* (also referred to as *VLAN translation* or *VLAN normalization*, depending on the switch vendor). VLAN mapping allows a trunk interface to change the 802.1Q tag. You must configure the switch with the mapping of one VLAN tag (used on the LAN-side of the Steelhead appliance) to another VLAN tag (used on the WAN-side of the Steelhead appliance) for packets to be sent or received. [Figure 7-25](#) shows multiple VLAN bridging with VLAN mapping deployment.

Figure 7-25. Multiple VLAN Bridging with VLAN Mapping



The following functionality makes it possible to optimize multiple VLANs with VLAN bridging:

- The VLAN mapping function on a switch changes the VLAN tags. A Steelhead appliance cannot do this.
- VLAN mapping takes 802.1Q tagged traffic from an incoming trunk switch-port and maps it to a different local VLAN.

For details on configuring your specific hardware, refer to the documentation provided with your switch.

CHAPTER 8 Virtual In-Path Deployments

This chapter describes virtual in-path deployments and summarizes the basic steps for configuring an in-path, load-balanced, Layer-4 switch deployment.

This chapter includes the following sections:

- [“Overview of Virtual In-Path Deployment” on page 217](#)
- [“Configuring an In-Path, Load-Balanced, Layer-4 Switch Deployment” on page 218](#)
- [“Configuring Flow Data Exports in Virtual In-Path Deployments” on page 220](#)

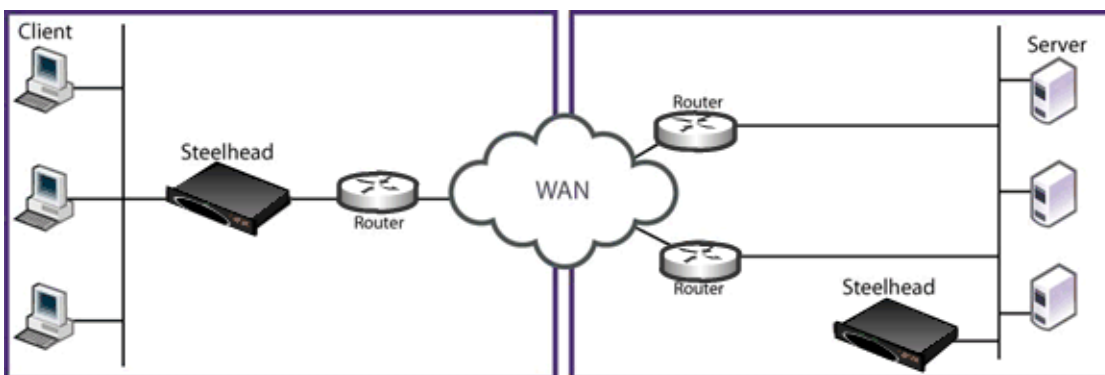
This chapter provides the basic steps for configuring one type of virtual in-path deployment. It does not provide detailed procedures for all virtual in-path deployments. Use this chapter as a general guide to virtual in-path deployments.

For details on the factors you must consider before you design and deploy the Steelhead appliance in a network environment, see [“Choosing the Right Steelhead Appliance” on page 16](#).

Overview of Virtual In-Path Deployment

In a virtual in-path deployment, the Steelhead appliance is virtually in the path between clients and servers. Traffic moves in and out of the same WAN interface, and the LAN interface is not used. This deployment differs from a physical in-path deployment in that a packet redirection mechanism directs packets to Steelhead appliances that are not in the physical path of the client or server.

Figure 8-1. Virtual In-Path Deployment on the Server-Side of the Network



Redirection mechanisms include:

- **Layer-4 switch** - You enable Layer-4 switch (or server load balancer) support when you have multiple Steelhead appliances in your network to manage large bandwidth requirements.

For details, see [“Configuring an In-Path, Load-Balanced, Layer-4 Switch Deployment” on page 218](#).

- **Hybrid** - A hybrid deployment is a deployment in which the Steelhead appliance is deployed either in a physical or virtual in-path mode, and has out-of-path mode enabled. A hybrid deployment is useful in which the Steelhead appliance must be referenced from remote sites as an out-of-path device (for example, to bypass intermediary Steelhead appliances).

For details, see [“Out-of-Path Deployments” on page 335](#).

- **PBR** - PBR enables you to redirect traffic to a Steelhead appliance that is configured as a virtual in-path device. PBR allows you to define policies that override routing behavior. For example, instead of routing a packet based on routing table information, it is routed based on the policy applied to the router. You define policies to redirect traffic to the Steelhead appliance and policies to avoid loop-back.

For details, see [“Policy-Based Routing Virtual In-Path Deployments” on page 255](#).

- **WCCP** - WCCP was originally implemented on Cisco routers, multilayer switches, and Web caches to redirect HTTP requests to local Web caches (Version 1). Version 2, which is supported on Steelhead appliances, can redirect any type of connection from multiple routers to multiple Web caches. For example, if you have multiple routers or if there is no in-path place for the Steelhead appliance, you can place the Steelhead appliance in a virtual in-path mode through the router so that they work together.

For details, see [“WCCP Virtual In-Path Deployments” on page 221](#).

- **Interceptor appliance** - The Interceptor appliance is a load balancer specifically used to distribute optimized traffic to a local cluster of Steelhead appliances. The Interceptor appliance is Steelhead appliance-aware, and so offers several benefits over other clustering techniques like WCCP and PBR. The Interceptor appliance is dedicated to redirecting packets for optimized connections to Steelhead appliances, but does not perform optimization itself. As a result, you can use the Interceptor appliance in extremely demanding network environments with extremely high throughput requirements. For details on the Interceptor appliance, see the *Interceptor Appliance Deployment Guide* and the *Interceptor Appliance User's Guide*.

For networks that contain firewalls or tunnels (VPN, GRE, IPSec transport mode) between Steelhead appliances and require manual tuning of the MTU values, see [“MTU Sizing” on page 445](#).

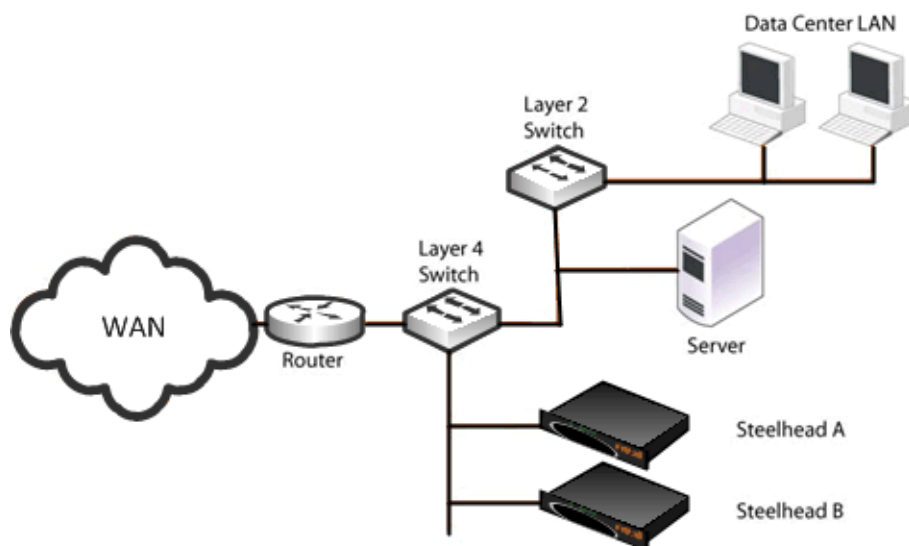
Configuring an In-Path, Load-Balanced, Layer-4 Switch Deployment

An in-path, load-balanced, Layer-4 switch deployment serves high-traffic environments or environments with large numbers of active TCP connections. It handles failures, scales easily, and supports all protocols.

When you configure the Steelhead appliance using a Layer-4 switch, you define the Steelhead appliances as a pool in which the Layer-4 switch redirects client and server traffic. Only one WAN interface on the Steelhead appliance is connected to the Layer-4 switch, and the Steelhead appliance is configured to send and receive data through that interface.

Figure 8-2 shows the server-side of the network where load balancing is required.

Figure 8-2. In-Path, Load-Balanced, Layer-4 Switch Deployment



To configure the client-side Steelhead appliance for in-path load-balanced, Layer-4 switch

- Configure the client-side Steelhead appliance as an in-path device. For details, see the *Steelhead Appliance Installation and Configuration Guide*.

To configure the server-side Steelhead appliance for in-path load-balanced, Layer-4 switch

1. Install and power on the Steelhead appliance.
2. Connect to the Steelhead appliance. Make sure you properly connect to the Layer-2 switch, for example:
 - On Steelhead A, plug the straight-through cable into the primary port of the Steelhead appliance and connect it to the LAN-side switch.
 - On Steelhead B, plug the straight-through cable into the primary port of the Steelhead appliance and connect it to the LAN-side switch.
3. Configure the Steelhead appliance in an in-path configuration.
4. Connect the Layer-4 switch to the Steelhead appliance:
 - On Steelhead A, plug the straight-through cable into the WAN port of the Steelhead appliance and the Layer-4 switch.
 - On Steelhead B, plug the straight-through cable into the WAN port of the Steelhead appliance and the Layer-4 switch.
5. Connect to the Management Console.
6. Go to the Configure > Optimization > General Service Settings page and enable Layer-4 switch support. For example, select Enable In-Path Support and Enable L4/PBR/WCCP Support.
7. Apply and save the new configuration in the Management Console.
8. Configure your Layer-4 switch as instructed by your switch documentation.

9. Go to the Configure > Maintenance > Services page and restart the optimization service.
10. View performance reports and system logs.
11. Perform the above steps for each Steelhead appliance in the cluster.

Configuring Flow Data Exports in Virtual In-Path Deployments

The Steelhead appliance supports the export of data flows to any compatible flow data collector. During data flow export, the flow data fields provide information such as the interface index that corresponds to the input and output traffic. An administrator can use the interface index to determine how much traffic is flowing from the LAN to the WAN and from the WAN to the LAN.

In virtual in-path deployments, such as the server-side of the network, traffic moves in and out of the same WAN interface; the LAN interface is not used. As a result, when the Steelhead appliance exports data to a flow data collector, all traffic has the WAN interface index. Though it is technically correct for all traffic to have the WAN interface index because the input and output interfaces are the same, this makes it impossible for an administrator to use the interface index to distinguish between LAN-to-WAN and WAN-to-LAN traffic.

In RiOS v6.0 or later, the fake index feature is enabled by default if you enable the *CascadeFlow* export option. Prior to RiOS v6.0, or if you are using standard NetFlow, you can work around this issue by turning on the Steelhead appliance fake index feature, which inserts the correct interface index before exporting data to a flow data collector. The fake index feature works only for optimized traffic, not unoptimized or passed-through traffic.

For details on how to configure fake index in the CLI in release prior to RiOS v6.0, see the appropriate version of the *Riverbed Command-Line Interface Reference Manual* or the *Steelhead Appliance Deployment Guide*.

Note: Subnet side rules are necessary for correct unoptimized or passed-through traffic reporting. For details, see the *Steelhead Appliance Management Console User's Guide*.

For details on exporting flow data, see [“Overview of Exporting Flow Data” on page 419](#).

CHAPTER 9 WCCP Virtual In-Path Deployments

This chapter describes how to configure WCCP to redirect traffic to one or more Steelhead appliances. This chapter includes the following sections:

- [“Overview of WCCP” on page 221](#)
- [“WCCP Fundamentals” on page 222](#)
- [“The Advantages and Disadvantages of WCCP” on page 228](#)
- [“Configuring WCCP” on page 229](#)
- [“Configuring Additional WCCP Features” on page 243](#)
- [“Flow Data in WCCP” on page 252](#)
- [“Verifying and Troubleshooting WCCP Configurations” on page 252](#)

This chapter provides basic information about WCCP network deployments and examples for configuring WCCP deployments. Use this chapter as a general guide to WCCP deployments.

For details on the factors you must consider before you design and deploy the Steelhead appliance in a network environment, see [“Choosing the Right Steelhead Appliance” on page 16](#).

For details on WCCP, see the Cisco documentation Web site at <http://www.cisco.com/en/US/docs/ios-xml/ios/ipapp/configuration/12-4/iap-wccp.html>.

Overview of WCCP

This section provides an overview of WCCP. WCCP Version 1 was originally implemented on Cisco routers, multilayer switches, and Web caches to redirect HTTP requests to local Web caches.

WCCP Version 2, which Steelhead appliances support, can redirect any type of connection from multiple routers to multiple WCCP clients (also referred to as *caches* or *engines*). Steelhead appliances deployed with WCCP can interoperate with remote Steelhead appliances deployed in any way, such as WCCP, PBR, in-path, and out-of-path.

WCCP requires either a Cisco router or a Cisco switch.

The most important factors in a successful WCCP implementation are the Cisco hardware platform and the IOS revision you use. There are many possible combinations of Cisco hardware and IOS revisions, and each combination has different capabilities.

Cisco platforms and IOS do not support all assignment methods, redirection methods, use of ACLs to control traffic, and interface interception directions. You can expect the Cisco minimum recommended IOS to change as WCCP becomes more widely used and new IOS technical issues are discovered.

Cisco recommends the following minimum IOS releases for specific hardware platforms.

Cisco Hardware	Cisco IOS
ASR 1000	2.2XE
ISR and 7200 Routers	12.1(14), 12.2(26), 12.3(13), 12.4(10), 12.1(3)T, 12.2(14)T, 12.3(14)T5, 12.4(15)T8, 12.4(24)T
ISR G2	15.0(1)M1
Catalyst 6500 with Sup720 or Sup32	12.2(18)SXF16, 12.2(33)SXI
Catalyst 6500 with Sup2	12.2(18)SXF16
Catalyst 7600	12.2(18)SXF16, 12.2(33)SXI
Catalyst 4500	12.2(46)SG
Catalyst 3750	12.2(46)SE
Nexus 7000	4.2.4

Regardless of how you configure a Steelhead appliance, if the Cisco IOS version on the router or switch is below the current Cisco minimum recommendations, it might be impossible to have a functioning WCCP implementation or the implementation might not have optimal performance.

WCCP Fundamentals

This section describes some of the fundamental concepts for configuring WCCP. This section includes the following topics:

- [“Service Groups” on page 222](#)
- [“Assignment Methods” on page 223](#)
- [“Redirection and Return Methods” on page 225](#)
- [“WCCP Clustering and Failover” on page 227](#)
- [“Multiple In-Path WCCP” on page 228](#)

Service Groups

A central concept of WCCP is the service group. The service group logically consists of up to 32 WCCP routers and 32 WCCP clients that work together to redirect and optimize traffic. WCCP routers are Cisco routers or switches capable of forwarding traffic meeting defined criteria. WCCP clients are the devices receiving this traffic. RiOS v6.1 or later includes multiple in-path WCCP (for details, see [“Multiple In-Path WCCP” on page 228](#)). The same WCCP routers and clients can participate in one or more service groups.

Service groups are differentiated by a service group number. The service group number is local to the site where WCCP is used. The service group number is not transmitted across the WAN.

When a router participates in a WCCP service group, it is configured to monitor traffic passing through a user-defined set of interfaces. When a router receives traffic of interest, it redirects the IP packets to be transmitted to a designated interface in another system in the WCCP service group.

Note: Riverbed recommends that you use WCCP service groups 61 and 62.

Routers redirect traffic to the Steelhead appliance interfaces in their WCCP service group. The assignment method and the load-balancing configuration determine which Steelhead appliance interface the router redirects traffic to.

Assignment Methods

This section describes WCCP assignment methods. This section includes the following topics:

- [“Hash Assignment” on page 223](#)
- [“Mask Assignment” on page 224](#)
- [“Choosing an Assignment Method” on page 225](#)

Routers participating in WCCP support two assignment methods. The assignment method affects how a router redirects traffic when multiple target systems are specified in a service group. Assignment methods are important when two or more Steelhead appliances are deployed at the same site for high availability or load balancing. The assignment methods are as follows:

- **Hash assignment** - Uses the software to calculate part of the load distribution, placing a significant load on the switch CPU.
- **Mask assignment** - Processes traffic entirely in hardware, so that the impact on the switch CPU is minimal. Mask assignment was specifically designed for hardware-based switches and routers.

Note: Do not confuse assignment methods with forwarding methods. Assignment methods determine how packets are distributed across multiple Steelhead appliances (through mask or hash), and forwarding methods determine how intercepted packets are forwarded from the router or switch to the Steelhead appliance (through GRE or Layer-2).

Hash Assignment

The hash assignment method redirects traffic based on a hashing scheme and the *weight* of the Steelhead appliance interfaces. A hashing scheme is a combination of the source IP address, destination IP address, source port, or destination port. The hash assignment method is commutative: a packet with a source IP address X, and a destination IP address Y, hashes to the same value as a packet with a source IP address Y, and a destination IP address X.

The weight of a Steelhead appliance is determined by the number of connections the Steelhead appliance supports. The default weight is based on the Steelhead appliance model number. The more connections a Steelhead appliance model supports, the heavier the weight of that model. You can modify the weight for each in-path interface to manually tune the proportion of traffic a Steelhead appliance interface receives.

The hash assignment method supports failover and load balancing. In a failover configuration, you configure one or more Steelhead appliance interfaces to be used only if no other Steelhead appliance interfaces within the WCCP service group are operating. To configure a Steelhead appliance interface for failover, set the Steelhead appliance interface weight to 0 on the desired service group.

If a Steelhead appliance interface has a weight of 0, and another Steelhead appliance interface in the same WCCP service group has a nonzero weight, the Steelhead appliance interface with the 0 weight does not receive redirected traffic. If all of the operating Steelhead appliances interfaces have a weight of 0, traffic is redirected equally among them.

Mask Assignment

The mask assignment method redirects traffic based on administrator-specified bits pulled, or masked, from the IP address and TCP port fields. Unlike the hash assignment method, these bits are not hashed. Instead, the Cisco switch concentrates the bits to construct an index into the load-balancing table.

You must carefully choose these bits. Mask assignment uses up to seven bits, which allows for a maximum of 128 buckets ($2^7=128$) for load balancing across Steelhead appliance interfaces in the same service group. RiOS v6.1 or later supports load balancing and redundancy per interface by configuring each Steelhead appliance interface with the appropriate service groups and router bindings (for details, see [“Multiple In-Path WCCP” on page 228](#)).

The mask assignment method processes the first packet for a connection in the router hardware. To force mask assignment, use the `assign-scheme` option for the **wccp service-group** CLI command:

```
wccp interface inpath0_0 service-group 61 routers 10.0.0.1 assign-scheme mask
```

Some Cisco platforms, such as the Catalyst 4500 and the Catalyst 3750, only support the mask assignment method.

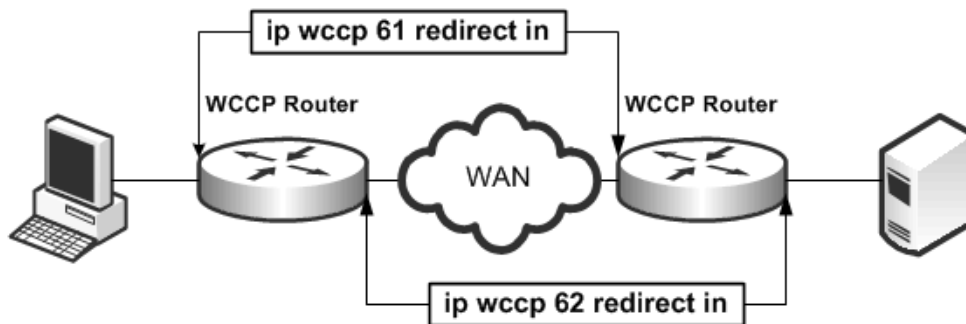
When you use the mask assignment method, you configure failover in the same manner as you do with the hash assignment method.

The mask assignment method requires that, for every connection, packets are redirected to the same Steelhead appliance or interface in both directions (client-to-server and server-to-client). To achieve redirection, you configure the following:

- Because only one set of masks can be used per service group, Riverbed recommends that you use two different service groups for inbound traffic from the client (group 61) and inbound traffic from the server (group 62).
- Configure the Cisco switch to redirect packets to a WCCP service group in the client-to-server direction, and to redirect packets to another WCCP group in the server-to-client direction. In most cases, service group 61 is placed on the inbound interface closest to the client and service group 62 is placed on the inbound interface closest to the server. Generally when you use mask, service group 61 is configured based on the source IP, and service group 62 is configured for the destination IP to maintain consistent assignment in either direction.

Figure 9-1 shows the reversed mask redirection technique.

Figure 9-1. Mask Assignment Method Packet Redirection



For details on mask assignment method parameters, see the *Riverbed Command-Line Interface Reference Manual*.

Choosing an Assignment Method

Unless otherwise specified in the Steelhead appliance interface WCCP service group setting, and if the router supports it, choose the hash assignment method. The hash assignment method generally achieves better load distribution, and mask assignment provides more user controlled configuration, including the ability to distribute traffic based on subnet mask values.

The following scenarios are instances when the mask assignment method is preferable:

- Certain lower-end Cisco switches (3750, 4000, 4500-series, among others) do not support hash assignment.
- The hash assignment method uses a NetFlow table entry on the switch for every connection. The NetFlow table entry can support up to 256K connections, depending on the hardware. However, when the switch runs out of NetFlow table entries, every WCCP-redirection packet is process-switched, which has a crippling effect on the switch CPU, and very large WCCP deployments are constrained to the mask assignment load distribution method.
- The hash assignment method processes the first packet of every new redirected connection using the switch CPU. The switch CPU installs the NetFlow table entry that is used to hardware-switch subsequent packets for a given connection. This process limits the number of connection setups a switch can perform per unit of time. Thus, in WCCP deployments where the connection setup rate is very high, the mask assignment method is the only option.
- In multiple Steelhead appliance environments, it is often desirable to send all users in subnet range to the same Steelhead appliance. Mask assignment provides a basic ability to leverage a branch subnet and Steelhead appliance to the same Steelhead appliance in a WCCP cluster.

Redirection and Return Methods

This section described the WCCP redirection and return methods. It includes the follow sections:

- [“WCCP Return Router Determination” on page 226](#)
- [“Best Practices for Determining a Redirection and Return Method” on page 227](#)

WCCP supports two methods for transmitting packets between a router or switch and Steelhead appliance interfaces: the GRE encapsulation method and the Layer-2 method. Steelhead appliances support both the Layer-2 and GRE encapsulation methods, in both directions, to and from the router or switch.

The Layer-2 method is generally preferred from a performance standpoint because it requires fewer resources from the router or switch than the GRE encapsulation does. The Layer-2 method modifies only the destination Ethernet address. However, not all combinations of Cisco hardware and IOS revisions support the Layer-2 method. Also, the Layer-2 method requires the absence of Layer-3 hops between the router or switch and the Steelhead appliance.

The GRE encapsulation method appends a GRE header to a packet before it is forwarded. This can cause fragmentation and imposes a performance penalty on the router and switch, especially during the GRE packet de-encapsulation process. This performance penalty can be too great for production deployments.

If your deployment requires the use of GRE return, you can configure the Steelhead appliance to automatically change the maximum segment size (MSS) for connections to 1432 bytes. In RiOS v6.0 and earlier, you must use the **in-path module wccp adjust-mss enable** command to avoid fragmentation due to the additional overhead of GRE encapsulation. In RiOS v6.1 or later, the default behavior is to change MSS to avoid fragmentation. You can overwrite this option with the command **no wccp adjust-mss enable**, although Riverbed does not recommend it.

You can avoid using the GRE encapsulation method for the traffic return path from the Steelhead appliance by using the Steelhead appliance **wccp override-return route-no-gre** or **wccp override-return sticky-no-gre** CLI commands. The **wccp override-return route-no-gre** command enables the Steelhead appliance to return traffic without GRE encapsulation to a Steelhead appliance in-path gateway, determined by the in-path routing table. The **wccp override-return sticky-no-gre** command enables the Steelhead appliance to return traffic without GRE encapsulation to the router that forwarded the traffic.

This occurs regardless of the method negotiated for returning traffic to the router or switch. Use the **wccp override-return route-no-gre** or **wccp override-return sticky-no-gre** commands only if the Steelhead appliance is no more than a Layer-2 hop away from the potential next-hop routers, and if the unencapsulated traffic does not pass through an interface that redirects the packet back to the Steelhead appliance (that is, there is no WCCP redirection loop). For details on the **wccp override-return route-no-gre** or **wccp override-return sticky-no-gre** commands, see the *Riverbed Command-Line Interface Reference Manual*.

The following table summarizes Cisco hardware platform support for redirection and return methods.

Cisco Hardware	Redirection and Return Method
Nexus 7000	Layer 2
ASR 1000	GRE or Layer 2
ISR and 7200 routers	GRE or Layer 2 (Layer 2 requires 12.4(20)T or later)
Catalyst 6500 with Sup720 or Sup32	GRE or Layer 2
Catalyst 6500 with Sup2	GRE or Layer 2
Catalyst 4500	Layer 2
Catalyst 3750	Layer 2

WCCP Return Router Determination

When a Steelhead appliance in a WCCP cluster transmits packets for optimized or pass-through connections, how it decides to address those packets depends on the RiOS version, WCCP configuration, and its in-path routing table. RiOS v6.0 or later includes more techniques to statefully track the originating router, both for Layer-2 and GRE methods.

Best Practices for Determining a Redirection and Return Method

Riverbed recommends the following best practices for determining your redirection and return method:

- Design your WCCP deployment so that your Steelhead appliances are no more than a Layer-2 hop away from the router or switch performing WCCP redirection.
- Do not configure a specific redirection or assignment method on your Steelhead appliance. Allow the Steelhead appliance to negotiate these settings with the router unless one of the reasons for using mask assignment applies to your deployment.

For details on mask assignment, see [“Choosing an Assignment Method” on page 225](#).

- Use the **wccp override-return route-no-gre** or **wccp override-return sticky-no-gre** commands only if the following conditions are both met:
 - The Steelhead appliance is no more than a Layer-2 hop away from the router or switch.
 - Unencapsulated traffic going to the next-hop router or switch does not pass through an interface that redirects the packet back to the Steelhead appliance (that is, there is no WCCP redirection loop). If this condition is not met, traffic redirected by the Steelhead appliance is continually returned to the same Steelhead appliance.

WCCP Clustering and Failover

WCCP clustering refers to two or more Steelhead appliances participating in the same service group. A single service group can support 32 WCCP clients, counting a single Steelhead appliance interface as a client. However, a single Steelhead appliance with multiple interfaces is typically not considered a cluster, even though each interface is considered a client.

Load balancing and redundancy is provided across all participating Steelhead appliances in a WCCP cluster by default. You can adjust the weight per each in-path interface to modify behavior for load balancing and redundancy. For example, you can configure a second in-path interface connected to a redundant device, with a smaller weight to be given a smaller proportion of traffic.

Certain timers in the WCCPv2 implementation directly affect failover time and are not adjustable. Every ten seconds, the Steelhead appliance and router exchange WCCP *Here I Am* and *I See You* messages (*Hello* messages). The router declares a client failed after missing three messages. This means that if a Steelhead appliance or Steelhead appliance interface fails, the router always waits between 20-30 seconds before declaring the Steelhead appliance down. If there are buckets (traffic distribution) assigned to the failed client, the router forwards traffic to the failed client during this interval, potentially black-holing that traffic. Traffic to other WCCP clients is redirected and optimized as normal. After the WCCP client is declared failed, buckets are recalculated. The buckets belonging to the failed device are distributed among the remaining WCCP clients. The hello and failure time intervals cannot be adjusted in the current WCCPv2 implementation.

In high-availability environments, optimization redundancy refers to the ability to quickly fail over to another appliance to continue application acceleration with minimal impact to users. However, no matter what deployment method is chosen, optimized connections to one Steelhead appliance are never statefully failed to a different Steelhead appliance. This is because the Steelhead appliances optimizing the connection act as TCP proxies and therefore are fully aware of the connection state. Although connection forwarding allows a Steelhead appliance to inform neighbors what connections are current, it does not exchange the full state of each connection, because this requires extensive and continuous updates with all neighbors. Without knowing the full state of a connection, a new device cannot safely resume the optimization of an existing connection. The majority of applications resend a SYN to establish a new connection with a redundant peer, transparent to the user. However, not all applications behave in this manner. Be aware that loss of a Steelhead appliance requires the client to reestablish a new TCP connection, which is usually transparent to the user.

In RiOS v6.5 or later, you must enable connection forwarding in a WCCP cluster. With connection forwarding enabled, the WCCP load balancing algorithm accounts for the total number of in-path interfaces of all neighbors in the service group when balancing the load across the interfaces. If you do not enable connection forwarding, the Steelhead appliance with the lowest IP address assigns all traffic flows to itself.

Multiple In-Path WCCP

RiOS v6.1 or later provides additional WCCP configuration, allowing each individual Steelhead appliance in-path interface to be configured as a WCCP client. Each configured in-path interface participates in WCCP service groups as an individual WCCP client, providing flexibility to determine load-balancing proportions and redundancy.

Prior to RiOS v6.1, WCCP was configured with the **wccp service group** command. This command now includes the interface as a mandatory parameter, making the command **wccp interface <interface> service group**.

For details, see *Riverbed Command-Line Interface Reference Manual*.

The Advantages and Disadvantages of WCCP

Physical in-path deployments require less initial and ongoing configuration and maintenance than out-of-path or virtual in-path deployments. This is because physical in-path Steelhead appliances are placed at the points in your network where data already flows. Thus, with in-path deployments you do not need to alter your existing network infrastructure.

For details on physical in-path deployments, see [“Physical In-Path Deployments” on page 171](#).

Virtual in-path techniques, such as WCCP, require more time to configure because the network infrastructure must be configured to redirect traffic to the Steelhead appliances.

WCCP also has the following advantages:

- **No rewiring required** - You do not need to move any wires during installation. At large sites with multiple active links, you can adjust wiring by moving individual links, one at a time, through the Steelhead appliances.
- **An option when no other is available** - At sites where a physical in-path deployment is not possible, WCCP might achieve the integration you need. For example, if your site has a WAN link terminating directly into a large access switch, there is no place to install a physical in-path Steelhead appliance.

WCCP has the following disadvantages:

- **Network design changes required** -WCCP deployments with multiple routers can require significant network changes (for example, spanning VLANs and GRE tunnels).
- **Hardware and IOS upgrades required** - To avoid hardware limitations and IOS issues, you must keep the Cisco platform and IOS revisions at the current minimum recommended levels. Otherwise, it might be *impossible* to create a stable deployment, regardless of how you configure the Steelhead appliance. For future IOS feature planning you must consider compatibility with WCCP.
- **Additional evaluation overhead** - It can take more time to evaluate the integration of the Steelhead appliances. This is in addition to evaluating Steelhead appliance performance gains. You might need Riverbed Professional Services to test and perform network infrastructure upgrades before any optimization can be performed. This is especially true when WCCP is deployed at numerous sites.

- **Additional configuration management** - You must create access control lists and manage them on an ongoing basis. At small sites, it might be feasible to redirect all traffic to the Steelhead appliances. However, at larger sites, access control lists might be required to ensure that traffic that cannot be optimized (for example, LAN-to-LAN traffic) is not sent to the Steelhead appliances.
- **GRE encapsulation** - If your network design does not support the presence of the Steelhead appliances and the Cisco router or switch interface in a common subnet, you must use GRE encapsulation for forwarding packets. Steelhead appliances can accommodate the subsequent extra performance utilization, but your existing router or switch might experience large resource utilization.

Configuring WCCP

This section describes how to configure WCCP and provides example deployments. This section includes the following topics:

- [“Basic Steps for Configuring WCCP” on page 229](#)
- [“Configuring a Simple WCCP Deployment” on page 230](#)
- [“Adding a Steelhead Appliance to an Existing WCCP Deployment” on page 233](#)
- [“Configuring a WCCP High Availability Deployment” on page 234](#)
- [“Configuring a Basic WCCP Router” on page 242](#)

Basic Steps for Configuring WCCP

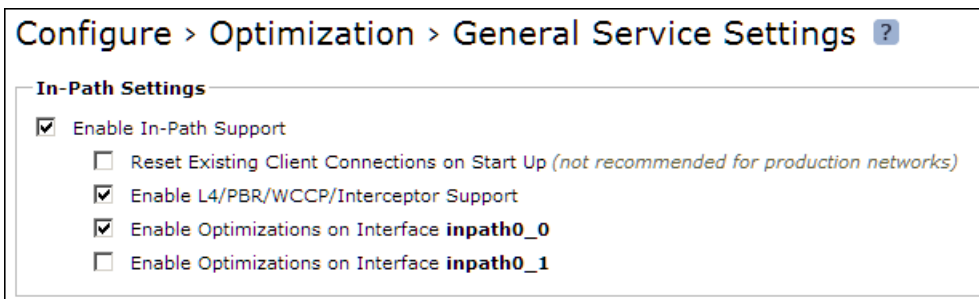
This section describes the basics steps to set up WCCP.

To perform the basic steps to configure WCCP

1. Configure the Steelhead appliance as an in-path device and enable in-path support.

You can use the CLI commands **in-path enable** and **in-path oop enable**, or the you can use In-Path Settings page shown in [Figure 9-2](#).

Figure 9-2. In-Path Settings



For details, see [“Physical In-Path Deployments” on page 171](#) and the *Steelhead Appliance Installation and Configuration Guide*.

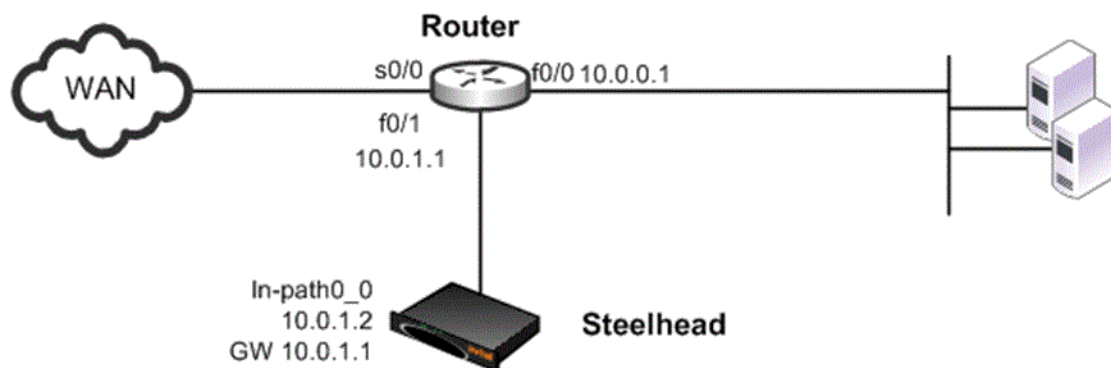
2. Enable WCCP on the router by creating a service group on the router.
3. Set the router to use WCCP to redirect traffic to the WCCP Steelhead appliance.

4. Attach the desired Steelhead appliance in-path interface WAN interface to the network. The WAN interface must be able to communicate with the switch or router on which WCCP is configured and where WCCP redirection takes place.
5. Add the service group on the WCCP Steelhead appliance interface.
6. Enable WCCP on the WCCP Steelhead appliance.

Configuring a Simple WCCP Deployment

Figure 9-3 shows a WCCP deployment that is simple to deploy and administer, and achieves high performance. This example includes a single router and a single Steelhead appliance.

Figure 9-3. A Single Steelhead Appliance and a Single Router



In this example:

- The router and the Steelhead appliance use WCCP service groups **61** and **62**. In this example, as long as the Steelhead appliance interface is a member of all of the service groups, and the service groups include all of the interfaces on all of the paths to and from the WAN, it does not matter whether a single service group or multiple service groups are configured.
- The Steelhead appliance wan0_0 interface is directly attached to the router, using a crossover cable.
- The Steelhead appliance virtual inpath0_0 interface uses the IP information that is visible to the router and the remote Steelhead appliances for data transfer.
- The Steelhead appliance does *not* have an encapsulation scheme in the WCCP service group configuration. Therefore, the Steelhead appliance informs the router that it supports both the GRE and the Layer-2 redirection methods. The method negotiated and used depends on the methods that the router supports.
- The Steelhead appliance default gateway return override is enabled with the **wccp override-return route-no-gre** command. Enabling this command decreases the resource utilization on the router. In this example, this is possible because returning packets do not match any subsequent WCCP interface redirect statements.

For details on the **wccp override-return route-no-gre** command, see [“Redirection and Return Methods” on page 225](#).

If you are using RiOS v4.x or earlier, see the following Riverbed Knowledge Base article, *What WCCP Redirect and Return Method Should I Use?*, located at <https://supportkb.riverbed.com/support/index?page=content&id=s15432>.

- The router uses the **ip wccp redirect exclude** CLI command on the router interface connected to the Steelhead appliance wan0_0 interface. This CLI command configures the router to never redirect packets arriving on this interface, even if they are later sent out of an interface with an **ip wccp redirect out** command. Although this is not required for this deployment, Riverbed recommends that you use it as a best practice.

Although the primary interface is not included in this example, Riverbed recommends that you connect the primary interface for management purposes. For details on configuring the primary interface, see the *Steelhead Appliance Management Console User's Guide*.

To configure WCCP on the Steelhead appliance

- On the Steelhead appliance, connect to the CLI and enter the following commands:

```
enable
configure terminal
#--- Configure the basic IP addressing of the Steelhead appliance
interface primary ip address 10.0.0.2 /24
ip default-gateway 10.0.0.1
interface inpath0_0 ip address 10.0.1.2 /24
ip in-path-gateway inpath0_0 10.0.1.1
in-path enable
#--- Enables virtual In-path support for WCCP / PBR / or Layer-4 switch
in-path oop enable
#--- Enable WCCP and create Service Groups 61 & 62; assign
#--- router IP addresses for each service group.
#--- If the Steelhead appliance is Layer-2 adjacent use the interface IP of the router
wccp enable
wccp interface inpath0_0 service-group 61 routers 10.0.1.1
wccp interface inpath0_0 service-group 62 routers 10.0.1.1
#--- If the router negotiates GRE return use route-no-gre to return
#--- the packets to the MAC of the next hop in the routing table instead
#--- of using GRE return. Alternately "wccp override-return sticky-no-gre"
#--- will return packets to the MAC address of the router that forwarded
#--- the packet to the Steelhead appliance.
wccp override-return route-no-gre
write memory
restart
```

Note: You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

In the following example, only traffic to or from IP addresses 192.168.1.1 is sent to the Steelhead appliance.

To configure WCCP on the Cisco router

- On the router, at the system prompt, enter the following commands:

```
enable
configure terminal
!--- Create the access control lists that determine what traffic to redirect
!--- to the Steelhead appliances. Creating two separate ACLs is optional
!--- Deny all traffic sourced from or destined to the Steelhead appliance
!--- in-path IP addresses and allow traffic from the client subnets to
!--- the server subnets
ip access-list extended wccp_acl_61
```

```

deny tcp 10.0.1.0 0.0.0.255 any
deny tcp any 10.0.1.0 0.0.0.255
permit tcp <client subnets> <server subnets>
!--- Deny all traffic sourced from or destined to the Steelhead appliance
!--- in-path IP addresses and allow traffic from the server subnets to
!--- the client subnets
ip access-list extended wccp acl 62
deny tcp 10.0.1.0 0.0.0.255 any
deny tcp any 10.0.1.0 0.0.0.255
permit tcp <server subnets> <client subnets>
!--- Enable WCCPv2 and service groups 61 & 62; define the redirect
!--- lists for each service group
ip wccp version 2
ip wccp 61 redirect-list wccp_acl_61
ip wccp 62 redirect-list wccp_acl_62
!--- Add WCCP service group 62 to the server facing interfaces
interface f0/0
    ip wccp 62 redirect in
!--- Add WCCP service group 61 to the client facing interfaces
interface s0/0
    ip wccp 61 redirect in
!--- As a best practice use "redirect exclude in" on the interfaces or VLANs
!--- that are connected to the Steelhead appliances. If you are using
!--- redirect out on any interface this command is REQUIRED.
interface f0/1
    ip wccp redirect exclude in
end
write memory

```

Tip: Enter configuration commands, one per line. End each command with CTRL-Z.

For details on how to verify the WCCP configuration, [“Verifying and Troubleshooting WCCP Configurations” on page 252](#).

Configuring WCCP using the mask assignment method is very similar to the standard WCCP configuration. The following example uses the mask of 0x3 that creates four buckets.

To configure WCCP on the Steelhead appliance using the mask assignment method

- On the Steelhead appliance, connect to the CLI and enter the following commands:

```

enable
configure terminal
!--- Configure the basic IP addressing of the Steelhead appliance
interface primary ip address 10.0.0.2 /24
ip default-gateway 10.0.0.1
interface inpath0_0 ip address 10.0.1.2 /24
ip in-path-gateway inpath0_0 10.0.1.1
in-path enable
!--- Enables virtual In-path support for WCCP / PBR / or L4 switch
in-path oop enable
!--- Enable WCCP and create Service Groups 61 & 62; assign
!--- router IP addresses for each service group.
!--- If the Steelhead appliance is L2 adjacent use the interface IP of the router
wccp enable
wccp interface inpath0_0 service-group 61 routers 10.0.1.1 assign-scheme mask src-ip-mask 0x3
wccp interface inpath0_0 service-group 62 routers 10.0.1.1 assign-scheme mask dst-ip-mask 0x3
!--- If the router negotiates GRE return use route-no-gre to return
!--- the packets to the MAC of the next hop in the routing table instead
!--- of using GRE return. Alternately "wccp override-return sticky-no-gre"
!--- will return packets to the MAC address of the router that forwarded
!--- the packet to the Steelhead appliance.

```

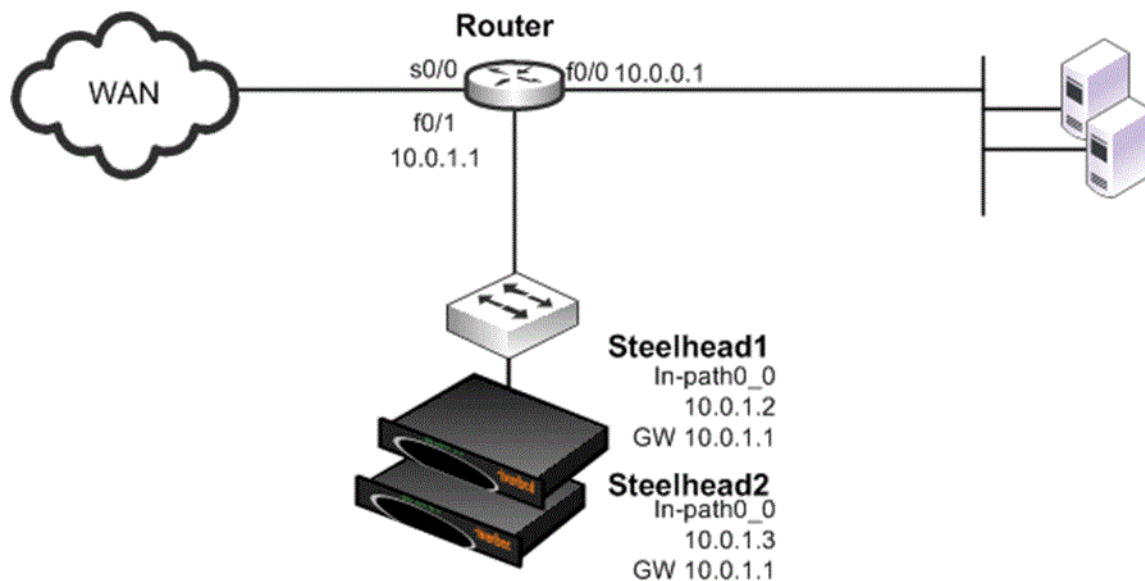
```
wccp override-return route-no-gre
write memory
restart
```

Adding a Steelhead Appliance to an Existing WCCP Deployment

You can have a maximum of 32 Steelhead appliances in your WCCP deployment. When you add new Steelhead appliances to an existing deployment, the buckets used by the router for load distribution are recalculated. New connections that were previously directed to one Steelhead appliance might be redirected, resulting initially in cold performance after you restart service.

Note: Adding a configuration to the existing Steelhead appliances requires a service restart during a performance window.

Figure 9-4. Adding a Steelhead Appliance to an Existing WCCP Deployment



To add an additional Steelhead appliance to an existing WCCP deployment

1. On Steelhead 1, connect to the CLI and enter the following commands:

```
enable
configure terminal
#--- Configure the basic IP addressing of the Steelhead appliance
interface inpath0_0 ip address 10.0.1.2 /24
ip in-path-gateway inpath0_0 10.0.1.1
in-path enable
#--- Enables virtual In-path support for WCCP / PBR / or Layer-4 switch
in-path oop enable
#--- Enable WCCP and create Service Groups 61 & 62; assign
#--- router IP addresses for each service group.
#--- If the Steelhead appliance is Layer-2 adjacent use the interface IP of the router
wccp enable
wccp interface inpath0_0 service-group 61 routers 10.0.1.1
wccp interface inpath0_0 service-group 62 routers 10.0.1.1
```

```
#--- If the router negotiates GRE return use route-no-gre to return
#--- the packets to the MAC of the next hop in the routing table instead
#--- of using GRE return. Alternately "wccp override-return sticky-no-gre"
#--- will return packets to the MAC address of the router that forwarded
#--- the packet to the Steelhead appliance.
wccp override-return route-no-gre
#--- Enables Connection Forwarding to neighbor 10.0.1.3
#--- allow-failure allows the Steelhead appliance to continue optimizing
#--- traffic even if the neighbor is down
steelhead communication enable
steelhead name Steelhead2 main-ip 10.0.1.3
steelhead communication allow-failure
steelhead communication advertiseresync
write memory
#--- Restart the optimization service
restart
```

2. On Steelhead 2, connect to the CLI and enter the following commands:

```
enable
configure terminal
#--- Configure the basic IP addressing of the Steelhead appliance
interface inpath0_0 ip address 10.0.1.3 /24
ip in-path-gateway inpath0_0 10.0.1.1
in-path enable
#--- Enables virtual In-path support for WCCP / PBR / or Layer-4 switch
in-path oop enable
#--- Enable WCCP and create Service Groups 61 & 62; assign
#--- router IP addresses for each service group.
#--- If the Steelhead appliance is Layer-2 adjacent use the interface IP of the router
wccp enable
wccp interface inpath0_0 service-group 61 routers 10.0.1.1
wccp interface inpath0_0 service-group 62 routers 10.0.1.1
#--- If the router negotiates GRE return use route-no-gre to return
#--- the packets to the MAC of the next hop in the routing table instead
#--- of using GRE return. Alternately "wccp override-return sticky-no-gre"
#--- will return packets to the MAC address of the router that forwarded
#--- the packet to the Steelhead appliance.
wccp override-return route-no-gre
#--- Enables Connection Forwarding to neighbor 10.0.1.2
#--- allow-failure allows the Steelhead appliance to continue optimizing
#--- traffic even if the neighbor is down
steelhead communication enable
steelhead name Steelhead1 main-ip 10.0.1.2
steelhead communication allow-failure
steelhead communication advertiseresync
write memory
#--- Restart the optimization service
restart
```

Configuring a WCCP High Availability Deployment

This section described configuring a WCCP high availability deployment. This section includes the following topics:

- [“Single Steelhead Appliance with Interface High Availability” on page 235](#)
- [“Dual WCCP Steelhead Appliances and Interfaces with High Availability” on page 238](#)

RiOS v6.1 or later supports redundancy across multiple interfaces. Previously, high availability was only available at the appliance level. The following examples show appliances running v6.1 or later with multiple WCCP interfaces. The same configuration concepts apply to versions before v6.1, except that each appliance can have only one WCCP interface configured.

If you use RiOS versions before v6.1 you cannot achieve the high availability shown in [“Single Steelhead Appliance with Interface High Availability,”](#) next. In [“Dual WCCP Steelhead Appliances and Interfaces with High Availability”](#) on page 238, you can provide appliance redundancy, but each Steelhead appliance does not have interface redundancy.

The examples in [“Single Steelhead Appliance with Interface High Availability”](#) on page 235 and [“Dual WCCP Steelhead Appliances and Interfaces with High Availability”](#) on page 238 show the configuration of Steelhead appliances for interface high availability. These examples focus solely on setting up multiple Steelhead appliance interfaces to communicate with multiple routers, and therefore omit any best practice recommendations on redirection and assignment method configurations.

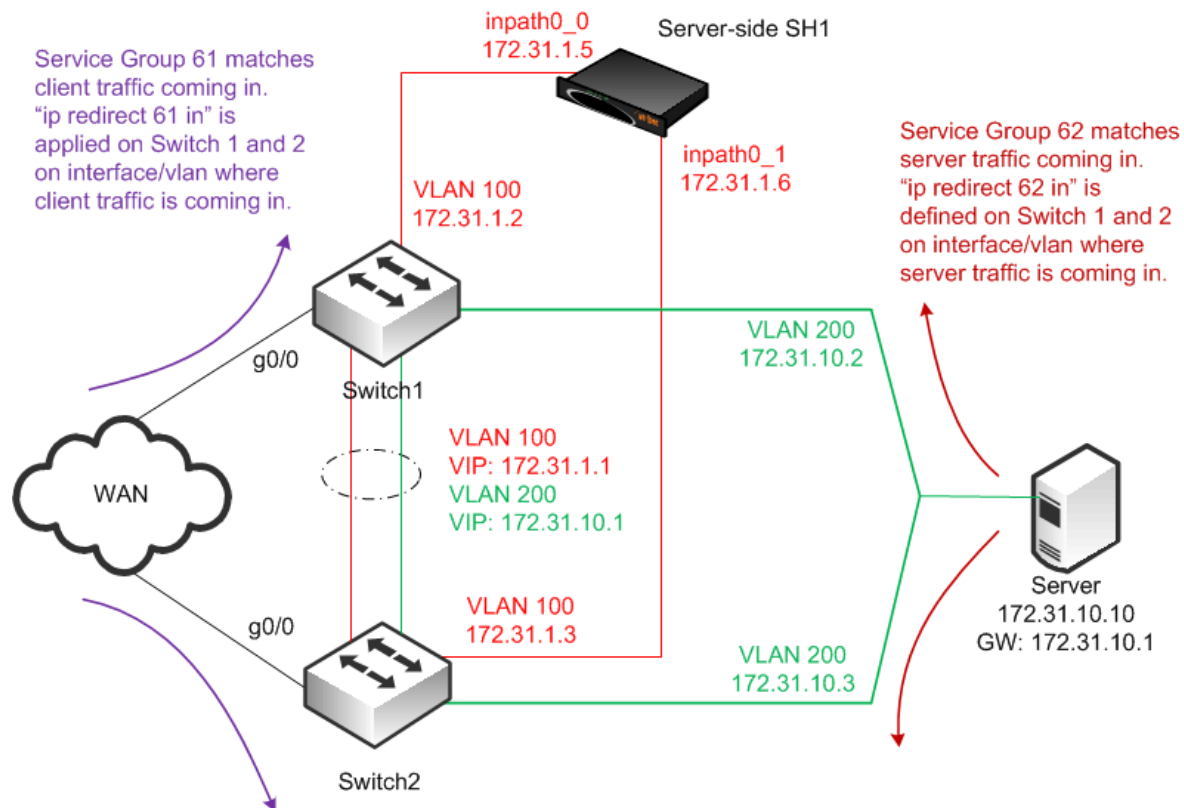
You must be familiar with [“Assignment Methods”](#) on page 223 and [“Redirection and Return Methods”](#) on page 225.

Single Steelhead Appliance with Interface High Availability

[Figure 9-5](#) shows a high availability WCCP deployment in which a single Steelhead appliance with two in-path interfaces and two routers are used in a WCCP configuration. This ensures that traffic continues to optimize in the event of a Steelhead appliance interface, router, or link failure. This example does not provide Steelhead appliance high availability.

Note: This deployment requires multiple in-path WCCP in RiOS v6.1 or later.

Figure 9-5. WCCP with Interface High Availability



In this example:

- The WCCP service groups are composed of two routers (Layer-3 switches) redirecting traffic and two Steelhead appliance interfaces acting as the cache engines. The Steelhead appliance is connected to both routers: wan0_0 goes to switch 1, and wan0_1 goes to switch 2.
- If a single Steelhead appliance interface fails, all traffic is forwarded to the remaining Steelhead appliance interface.

To configure WCCP with single Steelhead appliance with interface high availability

1. On the Steelhead 1, connect to the CLI and enter the following commands:

```
enable
configure terminal
#--- Configure the basic IP addressing of the Steelhead appliance. Primary address is used for
#--- management as well as for RiOS data store sync. The primary interface is not shown
#--- in the diagram as this can be attached to any accessible network.
interface primary ip address 10.10.1.10 /24
ip default-gateway 10.10.1.2
interface inpath0_0 ip address 172.31.1.5 /24
ip in-path-gateway inpath0_0 172.31.1.1
interface inpath0_1 ip address 172.31.1.6 /24
ip in-path-gateway inpath0_1 172.31.1.1
in-path enable
#--- Enables virtual In-path support for WCCP/PBR/ or Layer-4 switch
in-path oop enable
#--- Enable WCCP and create Service Groups 61 & 62; assign
#--- router IP addresses for each service group.
#--- If the Steelhead appliance is Layer-2 adjacent use the interface IP of the router
#--- If the Steelhead appliance is not Layer-2 adjacent use the RID (highest loopback) address
wccp enable
wccp interface inpath0_0 service-group 61 routers 172.31.1.2 172.31.1.3
wccp interface inpath0_0 service-group 62 routers 172.31.1.2 172.31.1.3
wccp interface inpath0_1 service-group 61 routers 172.31.1.2 172.31.1.3
wccp interface inpath0_1 service-group 62 routers 172.31.1.2 172.31.1.3
#--- The above omits configurations related to selecting redirection or assignment methods.
#--- It is recommended to read, understand, and select the methods most appropriate for the
#--- environment. For example, the majority of L3 switches prefer L2 redirection and mask
#--- assignment. When using mask assignment, follow the best practices to ensure consistent
#--- assignment in either direction, typically by using source IP mask in one service group,
#--- and destination IP mask in the other.
wccp override-return sticky-no-gre
write memory
restart
```

Note: You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

2. To configure WCCP on Cisco router 1 (Switch 1), at the system prompt, enter the following commands:

```
enable
configure terminal
!--- Create the access control lists that determine what traffic to redirect
!--- to the Steelhead appliances. Creating two separate ACLs is optional
!--- Deny all traffic sourced from or destined to the Steelhead appliance
!--- in-path IP addresses and allow traffic from the client subnets to
!--- the server subnets
ip access-list extended wccp_acl_61
deny tcp 172.31.1.0.0.0.255 any
```

```

deny tcp any 172.31.1.0 0.0.0.255
permit tcp <client subnets> <server subnets>
!--- Deny all traffic sourced from or destined to the Steelhead appliance
!--- in-path IP addresses and allow traffic from the server subnets to
!--- the client subnets
ip access-list extended wccp_acl_62
deny tcp 172.31.1.0.0.0.255 any
deny tcp any 172.31.1.0 0.0.0.255
permit tcp <server subnets> <client subnets>
!--- Enable WCCPv2 and service groups 61 & 62; define the redirect
!--- lists for each service group
ip wccp version 2
ip wccp 61 redirect-list wccp_acl_61
ip wccp 62 redirect-list wccp_acl_62
!--- As a best practice use "redirect exclude in" on the interfaces or VLANs
!--- that are connected to the Steelhead appliances. If you are using
!--- redirect out on any interface this command is REQUIRED.
interface vlan 100
    ip wccp redirect exclude in
!--- Add WCCP service group 61 to the client facing interfaces, in this example
!--- clients traffic arrives via gigabit interface 0/0
interface g0
    ip wccp 61 redirect in
!--- Add WCCP service group 62 to the server facing interfaces, in this example
!--- servers are coming in via the LAN on VLAN 200
interface vlan 200
    ip wccp 62 redirect in
end
write memory

```

Tip: Enter configuration commands, one per line. End each command with CTRL-Z.

3. To configure WCCP on Cisco router 2 (Switch 2), at the system prompt, enter the following commands:

```

enable
configure terminal
!--- Deny all traffic sourced from or destined to the Steelhead appliance
!--- in-path IP addresses and allow traffic from the client subnets to
!--- the server subnets
ip access-list extended wccp_acl_61
deny tcp 172.31.1.0.0.0.255 any
deny tcp any 172.31.1.0 0.0.0.255
permit tcp <client subnets> <server subnets>
!--- Deny all traffic sourced from or destined to the Steelhead appliance
!--- in-path IP addresses and allow traffic from the server subnets to
!--- the client subnets
ip access-list extended wccp_acl_62
deny tcp 172.31.1.0.0.0.255 any
deny tcp any 172.31.1.0 0.0.0.255
permit tcp <server subnets> <client subnets>
!--- Enable WCCPv2 and service groups 61 & 62; define the redirect
!--- lists for each service group
ip wccp version 2
ip wccp 61 redirect-list wccp_acl_61
ip wccp 62 redirect-list wccp_acl_62
!--- As a best practice use "redirect exclude in" on the interfaces or VLANs
!--- that are connected to the Steelhead appliances. If you are using
!--- redirect out on any interface this command is REQUIRED.
interface vlan 100
    ip wccp redirect exclude in
!--- Add WCCP service group 61 to the client facing interfaces, in this example
!--- client traffic arrives via gigabit interface 0/0.

```

```

interface g0/0
  ip wccp 61 redirect in
!--- Add WCCP service group 62 to the server facing interfaces, in this example
!--- servers are coming in via the LAN on VLAN 200
interface vlan 200
  ip wccp 62 redirect in
end
write memory

```

Tip: Enter configuration commands, one per line. End each command with CTRL-Z.

For details on verifying the WCCP configuration, see [“Verifying and Troubleshooting WCCP Configurations” on page 252](#).

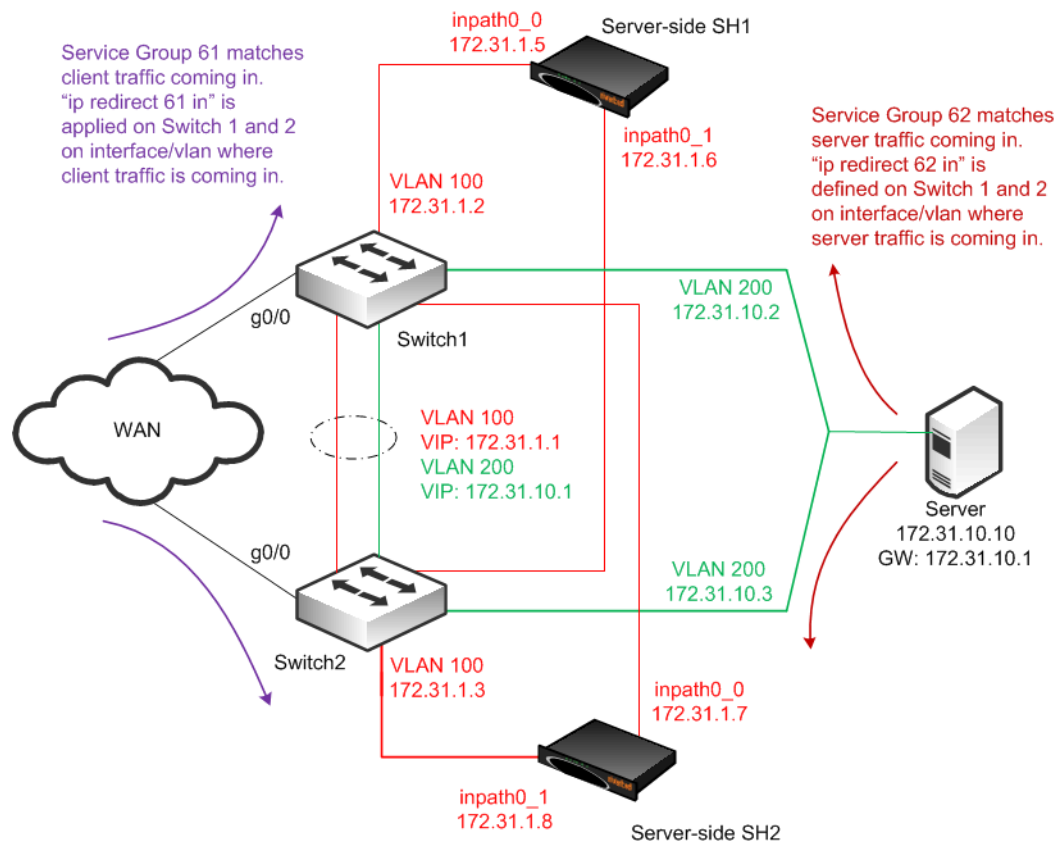
Dual WCCP Steelhead Appliances and Interfaces with High Availability

Figure 9-6 shows a high availability WCCP deployment where two Steelhead appliances with two in-path interfaces and two routers are used in a WCCP configuration. This ensures that traffic continues to be optimized in the event of a Steelhead appliance interface or router failure.

This deployment requires multiple in-path WCCP in RiOS v6.1 or later.

RiOS data store synchronization is commonly used in high availability designs. You can configure RiOS data store synchronization between any two local Steelhead appliances, regardless of how they are deployed: physical in-path, virtual in-path, or out-of-path. For details on data store synchronization, see [“RiOS Data Store Synchronization” on page 15](#).

Figure 9-6. High Availability WCCP with RiOS Data Store Synchronization



In this example:

- The Steelhead appliances are connected to both routers (Layer-3 switches). For each Steelhead appliance, wan0_0 is connected to Switch 1, and wan0_1 is connected to Switch 2.
- The WCCP service groups are composed of two routers redirecting traffic and four Steelhead appliance interfaces acting as the cache engines.
- If a single Steelhead appliance interface fails, all traffic is forwarded to the remaining Steelhead appliance interfaces, including the second interface on the same Steelhead appliance.
- If a single Steelhead appliance fails, all traffic is forwarded to the other Steelhead appliance's two in-path interfaces.
- Because the two Steelhead appliances synchronize their RiOS data stores, the remaining Steelhead appliance provides the same level of acceleration as the failed Steelhead appliance.

If you are using a cluster of WCCP-attached Steelhead appliances, all remote client-side Steelhead appliances must have probe caching disabled using the **no in-path probe-caching enable** command.

To configure dual WCCP Steelhead appliances with interfaces with high availability

1. To configure WCCP on Steelhead1, connect to the CLI and enter the following commands:

```
enable
configure terminal
#--- Configure the basic IP addressing of the Steelhead appliance.
#--- Primary address is used for management as well as for RiOS data store sync.
#--- The primary interface is not shown in the diagram
#--- as this can be attached to any accessible network.
interface primary ip address 10.10.1.10 /24
ip default-gateway 10.10.1.2
interface inpath0_0 ip address 172.31.1.5 /24
ip in-path-gateway inpath0_0 172.31.1.1
interface inpath0_1 ip address 172.31.1.6 /24
ip in-path-gateway inpath0_1 172.31.1.1
in-path enable
#--- Enables virtual In-path support for WCCP/PBR/ or Layer-4 switch
in-path oop enable
#--- Enables Connection Forwarding to neighbor 172.31.1.7
#--- allow-failure allows the Steelhead appliance to continue optimizing
#--- traffic even if the neighbor is down
steelhead communication enable
steelhead name SH2 main-ip 172.31.1.7
steelhead communication allow-failure
steelhead communication advertiseressync
#--- Enable WCCP and create Service Groups 61 & 62; assign
#--- router IP addresses for each service group.
#--- If the Steelhead appliance is Layer-2 adjacent use the interface IP of the router
#--- If the Steelhead appliance is not Layer-2 adjacent use the RID (highest loopback) address
wccp enable
wccp interface inpath0_0 service-group 61 routers 172.31.1.2 172.31.1.3
wccp interface inpath0_0 service-group 62 routers 172.31.1.2 172.31.1.3
wccp interface inpath0_1 service-group 61 routers 172.31.1.2 172.31.1.3
wccp interface inpath0_1 service-group 62 routers 172.31.1.2 172.31.1.3
#--- The above omits configurations related to selecting redirection or assignment methods.
#--- It is recommended to read, understand, and select the methods most appropriate for the
#--- environment. For example, the majority of L3 switches prefer L2 redirection and mask
#--- assignment. When using mask assignment, follow the best practices to ensure consistent
#--- assignment in either direction, typically by using source IP mask in one service group,
#--- and destination IP mask in the other.
wccp override-return sticky-no-gre
#--- Enable RiOS data store synchronization and set this Steelhead appliance as the master
datastore sync master
```

```

datastore sync peer-ip 10.10.1.13
datastore sync enable
write memory
restart

```

Note: You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

2. To configure WCCP on Steelhead2, connect to the CLI and enter the following commands:

```

enable
configure terminal
#--- Configure the basic IP addressing of the Steelhead appliance.
#--- Primary address is used for management as well as for RiOS data store sync.
#--- The primary interface is not shown in the diagram as this
#--- can be attached to any accessible network.interface primary ip address 10.10.1.13 /24
interface primary ip address 10.1.1.13 /24
ip default-gateway 10.10.1.3
interface inpath0_0 ip address 172.31.1.7 /24
ip in-path-gateway inpath0_0 172.31.1.1
interface inpath0_1 ip address 172.31.1.8 /24
ip in-path-gateway inpath0_1 172.31.1.1
in-path enable
#--- Enables virtual In-path support for WCCP / PBR / or Layer-4 switch
in-path oop enable
#--- Enables Connection Forwarding to neighbor 172.31.1.5
#--- allow-failure allows the Steelhead appliance to continue optimizing
#--- traffic even if the neighbor is down
steelhead communication enable
steelhead name SH1 main-ip 172.31.1.5
steelhead communication allow-failure
steelhead communication advertiseressync
#--- Enable WCCP and create Service Groups 61 & 62; assign
#--- router IP addresses for each service group.
#--- If the Steelhead appliance is Layer-2 adjacent use the interface IP of the router
#--- If the Steelhead appliance is not Layer-2 adjacent use the RID (highest loopback) address
wccp enable
wccp interface inpath0_0 service-group 61 routers 172.31.1.2 172.31.1.3
wccp interface inpath0_0 service-group 62 routers 172.31.1.2 172.31.1.3
wccp interface inpath0_1 service-group 61 routers 172.31.1.2 172.31.1.3
wccp interface inpath0_1 service-group 62 routers 172.31.1.2 172.31.1.3
#--- The above omits configurations related to selecting redirection or assignment methods.
#--- It is recommended to read, understand, and select the methods most appropriate for the
#--- environment. For example, the majority of L3 switches prefer L2 redirection and mask
#--- assignment. When using mask assignment, follow the best practices to ensure consistent
#--- assignment in either direction, typically by using source IP mask in one service group,
#--- and destination IP mask in the other.
wccp override-return sticky-no-gre
#--- Enables RiOS data store synchronization and sets this Steelhead appliance as the slave
no datastore sync master
datastore sync peer-ip 10.10.1.10
datastore sync enable
write memory
restart

```

Note: You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

3. To configure WCCP on Cisco router 1 (Switch 1), at the system prompt, enter the following commands:

```
enable
configure terminal
!--- Deny all traffic sourced from or destined to the Steelhead appliance
!--- in-path IP addresses and allow traffic from the client subnets to
!--- the server subnets
ip access-list extended wccp_acl_61
deny tcp 172.31.1.0.0.0.255 any
deny tcp any 172.31.1.0 0.0.0.255
permit tcp <client subnets> <server subnets>
!--- Deny all traffic sourced from or destined to the Steelhead appliance
!--- in-path IP addresses and allow traffic from the server subnets to
!--- the client subnets
ip access-list extended wccp_acl_62
deny tcp 172.31.1.0.0.0.255 any
deny tcp any 172.31.1.0 0.0.0.255
permit tcp <server subnets> <client subnets>
!--- Enable WCCPv2 and service groups 61 & 62; define the redirect
!--- lists for each service group
ip wccp version 2
ip wccp 61 redirect-list wccp_acl_61
ip wccp 62 redirect-list wccp_acl_62
!--- As a best practice use "redirect exclude in" on the interfaces or VLANs
!--- that are connected to the Steelhead appliances. If you are using
!--- redirect out on any interface this command is REQUIRED.
interface vlan 100
    ip wccp redirect exclude in
!--- Add WCCP service group 61 to the client facing interfaces, in this example
!--- client traffic arrives via gigabit interface 0/0.
interface g0/0
    ip wccp 61 redirect in
!--- Add WCCP service group 62 to the server facing interfaces, in this example
!--- servers are coming in via the LAN on VLAN 200
interface vlan 200
    ip wccp 62 redirect in
end
write memory
```

Tip: Enter configuration commands, one per line. End each command with CTRL-Z.

4. To configure WCCP on Cisco router 2 (Switch 2), at the system prompt, enter the following commands:

```
enable
configure terminal
!--- Deny all traffic sourced from or destined to the Steelhead appliance
!--- in-path IP addresses and allow traffic from the client subnets to
!--- the server subnets
ip access-list extended wccp_acl_61
deny tcp 172.31.1.0.0.0.255 any
deny tcp any 172.31.1.0 0.0.0.255
permit tcp <client subnets> <server subnets>
!--- Deny all traffic sourced from or destined to the Steelhead appliance
!--- in-path IP addresses and allow traffic from the server subnets to
!--- the client subnets
ip access-list extended wccp_acl_62
deny tcp 172.31.1.0.0.0.255 any
deny tcp any 172.31.1.0 0.0.0.255
permit tcp <server subnets> <client subnets>
ip wccp version 2
ip wccp 61 redirect-list wccp_acl_61
ip wccp 62 redirect-list wccp_acl_62
```

```

!--- As a best practice use "redirect exclude in" on the interfaces or VLANs
!--- that are connected to the Steelhead appliances. If you are using
!--- redirect out on any interface this command is REQUIRED.
interface vlan 100
    ip wccp redirect exclude in
!--- Add WCCP service group 61 to the client facing interfaces, in this example
!--- client traffic arrives via gigabit interface 0/0.
interface g0/0
    ip wccp 61 redirect in
!--- Add WCCP service group 62 to the server facing interfaces, in this example
!--- servers are coming in via the LAN on VLAN 200
interface vlan 200
    ip wccp 62 redirect in
end
write memory

```

Tip: Enter configuration commands, one per line. End each command with CTRL-Z.

For details on how to verify the WCCP configuration, [“Verifying and Troubleshooting WCCP Configurations” on page 252](#).

Configuring a Basic WCCP Router

This section summarizes some of the basic WCCP router configuration commands. For complete details on WCCP router configuration commands, refer to your router documentation.

To enable WCCP and define a service group on the router

- On the router, at the system prompt, enter the following commands:

```

enable
configure terminal
ip wccp <service_group> <redirect list>
end
write memory

```

Important: The service group you specify on the router must also be set on the WCCP Steelhead appliance.

Note: The WCCP protocol allows you to add up to 32 Steelhead appliances and 32 routers to a service group.

To specify inbound traffic redirection for each router interface

- On the router, at the system prompt, enter the following commands:

```

enable
configure terminal
!--- Add WCCP service group 61 to the client-facing interfaces
interface FastEthernet 0/0
ip wccp 61 redirect in
!--- Add WCCP service group 62 to the server-facing interfaces
interface serial 0
ip wccp 62 redirect in
end
write memory

```

The **ip wccp [SR]** router command is not additive. After you run the **end** and **write memory** options for an **ip wccp [SR]** command, you cannot use another **ip wccp [SR]** command to augment information you previously specified. This is not true with the Steelhead appliance **wccp interface** command.

To retain information you previously specified with **ip wccp [SR]**, you must enter a new **ip wccp** command that includes the information you previously specified, and whatever you want to configure.

For example, you can configure your router using the following set of commands:

```
enable
configure terminal
ip wccp 61 redirect-list 100
end
write memory
```

If you want to specify a password on the router later, the command **ip wccp 61 password <your_password>** overwrites the previous redirect list configuration.

To retain the previous redirect list configuration and set a password, you must use the following command:

```
ip wccp 61 redirect-list 100 password <your_password>
```

For example:

```
enable
configure terminal
ip wccp 61 redirect-list 100 password XXXYYZZ
end
write memory
```

Configuring Additional WCCP Features

This section describes additional WCCP features and how to configure them. This section includes the following topics:

- [“Specifying the Service Group Password” on page 243](#)
- [“Configuring Multicast Groups” on page 244](#)
- [“Configuring Group Lists to Limit Service Group Members” on page 245](#)
- [“Configuring Access Control Lists” on page 246](#)
- [“Configuring Load Balancing in WCCP” on page 249](#)

Specifying the Service Group Password

You can configure password authentication of WCCP protocol messages between the router and the Steelhead appliance interface:

- The router service group must match the service group password configured on the WCCP Steelhead appliance interface.
- The same password must be configured on the router and the WCCP Steelhead appliance interface.
- Passwords must be no more than eight characters.

Important: The following router commands are not required for the example network configurations in this chapter. Use caution when you enter the **ip wccp [SG]** router command because each **ip wccp [SG]** router command overwrites the previous **ip wccp [SG]** router command. You cannot use an **ip wccp [SG]** router command to augment **ip wccp [SG]** router commands you previously issued. For details, see [“Configuring a Basic WCCP Router” on page 242](#).

To specify the service group password on the WCCP router

- On the router, at the system prompt, enter the following commands:

```
enable
configure terminal
ip wccp <service_group> password <your_password>
end
write memory
```

Tip: Enter configuration commands, one per line. End each command with CTRL-Z.

To set the service group password on the WCCP Steelhead appliance interface

- Connect to the Riverbed CLI on the WCCP Steelhead appliance and enter the following commands:

```
enable
configure terminal
wccp interface <interface> service-group <service-id> routers <IP address> password
<your_password>
write memory
restart
```

For example, to set the password on inpath0_0, where the router service group is **61** and the router IP address is 10.1.0.1, enter the following command:

```
wccp inpath0_0 service-group 61 routers 10.1.0.1 password XXXYYYYZ
```

Note: You must set the same password on the Steelhead appliance interface and the Cisco router.

Note: You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

Configuring Multicast Groups

If you add multiple routers and Steelhead appliance interfaces to a service group, you can configure them to exchange WCCP protocol messages through a multicast group.

Configuring a multicast group is advantageous because if a new router is added, it does not need to be explicitly added on each Steelhead appliance interface.

Important: The following router commands are not required for the example network configurations in this chapter. Use caution when you enter the **ip wccp [SG]** router command because each **ip wccp [SG]** router command overwrites the previous **ip wccp [SG]** router command. You cannot use an **ip wccp [SG]** router command to augment **ip wccp [SG]** router commands you previously issued. For details, see [“Configuring a Basic WCCP Router” on page 242](#).

To configure multicast groups on the WCCP router

- On the router, at the system prompt, enter the following commands:

```
enable
configure terminal
ip wccp 61 group-address 239.0.0.1
interface fastEthernet 0/0
ip wccp 61 redirect in
ip wccp 61 group-listen
end
write memory
```

Tip: Enter configuration commands, one per line. End each command with CTRL-Z.

Note: Multicast addresses must be between 224.0.0.0 and 239.255.255.255.

To configure multicast groups on the WCCP Steelhead appliance interface

- Connect to the Riverbed CLI on the WCCP Steelhead appliance and enter the following commands:

```
enable
configure terminal
wccp enable
wccp mcast-ttl 10
wccp interface inpath0_0 service-group 61 routers 239.0.0.1
write memory
restart
```

Note: You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

Note: You must set the same password on the Steelhead appliance interface and the Cisco router.

Configuring Group Lists to Limit Service Group Members

You can configure a group list on your router to limit service group members (for instance, Steelhead appliance interfaces) by IP address.

For example, if you want to allow only Steelhead appliance interfaces with IP addresses 10.1.1.23 and 10.1.1.24 to join the router service group, you create a group list on the router.

Important: The following router command is not required for the example network configurations in this chapter. Use caution when you enter the **ip wccp [SG]** router command because each **ip wccp [SG]** router command overwrites the previous **ip wccp [SG]** router command. You cannot use an **ip wccp [SG]** router command to augment **ip wccp [SG]** router commands you previously issued. For details, see [“Configuring a Basic WCCP Router” on page 242](#).

To configure a WCCP router group list

- On the WCCP router, at the system prompt, enter the following commands:

```
enable
configure terminal
access-list 1 permit 10.1.1.23
access-list 1 permit 10.1.1.24
ip wccp 61 group-list 1
interface fastEthernet 0/0
ip wccp 61 redirect in
end
write memory
```

Tip: Enter configuration commands, one per line. End each command with CTRL-Z.

Configuring Access Control Lists

This section describes how to configure access control lists (ACLs). This section includes the following topics:

- [“Using Access Control Lists for Specific Traffic Redirection” on page 246](#)
- [“Cisco Access Control List Command Parameters” on page 247](#)
- [“Using Access Control Lists with WCCP” on page 248](#)

When you configure ACLs, consider the following:

- ACLs are processed in order, from top to bottom. As soon as a particular packet matches a statement, it is processed according to that statement and the packet is not evaluated against subsequent statements. The order of your access control list statements is very important.
- If port information is not explicitly defined, all ports are assumed.
- By default, all lists include an implied **deny all** Cisco command at the end, which ensures that traffic that is not explicitly included is denied. You cannot change or delete this implied entry.

Using Access Control Lists for Specific Traffic Redirection

If redirection is based on traffic characteristics other than ports, you can use ACLs on the router to define which traffic is redirected.

If you only want the traffic for IP address 10.2.0.0/16 to be redirected to the WCCP Steelhead appliance, configure the router according to the following example.

Important: The following router command is not required for the example network configurations in this chapter. Use caution when you enter the **ip wccp [SG]** router command because each **ip wccp [SG]** router command overwrites the previous **ip wccp [SG]** router command. You cannot use an **ip wccp [SG]** router command to augment **ip wccp [SG]** router commands you previously issued. For details, see [“Configuring a Basic WCCP Router” on page 242](#).

To configure specific traffic redirection on the router

- On the router, at the system prompt, enter the following commands:

```
enable
configure terminal
access-list 101 permit tcp any 10.2.0.0 0.0.255.255
```



```

access-list 101 permit tcp 10.2.0.0 0.0.255.255 any
ip wccp 61 redirect-list 101
interface fastEthernet 0/0
ip wccp 61 redirect in
end
interface serial0
ip wccp 61 redirect in
end
write memory

```

Important: If you have defined fixed-target rules, redirect traffic in one direction, as shown this example.

Tip: Enter configuration commands, one per line. End each command with CTRL-Z.

Cisco Access Control List Command Parameters

This section describes the Cisco **access-list** router command for using ACLs to configure WCCP redirect lists. For details on ACL commands, refer to your router documentation.

The **access-list** router command has the following syntax:

access-list <access_list_number> [permit | deny] tcp <source IP/mask> <source_port> <destination IP/mask> <destination_port>

access_list_number	Specifies the number from 1-199 that identifies the access control list. Standard access control lists are numbered 1-99; extended access control lists are numbered 100-199. A standard access control list matches traffic based on source IP address. An extended access control list matches traffic based on source or destination IP address. Riverbed recommends that you use extended IP access control lists.
permit deny	Specifies whether the redirect list allows or stops traffic redirection. Specify permit to allow traffic redirection; specify deny to stop traffic redirection.
tcp	Specifies the traffic to redirect. WCCP only redirects TCP traffic. Use this option only when configuring a redirect list for WCCP.
source IP/mask	Specifies the source IP address and mask. To set the mask, specify 0 or 1, where 0 = match and 1 = does not matter, for example: <ul style="list-style-type: none"> • any - Matches any IP address. • 10.1.1.0 0.0.0.255 - Matches any host on the 10.1.1.0 network. • 10.1.1.1 0.0.0.0 - Matches host 10.1.1.1 exactly. • 10.1.1.1 - Matches host 10.1.1.1 exactly. This option is identical to specifying 10.1.1.1 0.0.0.0.
source_port	Specifies the source port number or corresponding keyword, for example: <ul style="list-style-type: none"> • eq 80 or eq www - Identical options that match port 80. • gt 1024 - Matches any port greater than 1024. • lt 1024 - Matches any port less than 1024. • neq 80 - Matches any port except port 80. • range 80 90 - Matches any port between and including 80 through 90. Cisco routers support many keywords. For details, refer to your router documentation.

destination IP/mask	<p>Specifies the destination IP address and mask. To set the mask, specify 0 or 1, where 0 = match and 1 = does not matter, for example:</p> <ul style="list-style-type: none"> • any - Matches any IP address. • 10.1.1.0 0.0.0.255 - Matches any host on the 10.1.1.0 network. • 10.1.1.1 0.0.0.0 - Matches host 10.1.1.1 exactly. • 10.1.1.1 - Matches host 10.1.1.1 exactly. This option is identical to specifying 10.1.1.1 0.0.0.0.
destination_port	<p>Specifies the destination port number or corresponding keyword, for example:</p> <ul style="list-style-type: none"> • eq 80 or eq www - Identical options that match port 80. • gt 1024 - Matches any port greater than 1024. • lt 1024 - Matches any port less than 1024. • neq 80 - Matches any port except port 80. • range 80 90 - Matches any port between and including 80 through 90. <p>Cisco routers support many keywords. For details, refer to your router documentation.</p>

Using Access Control Lists with WCCP

To avoid requiring the router to do extra work, Riverbed recommends that you create an ACL that routes only traffic that you intend to optimize to the Steelhead appliance.

Suppose your network is structured so that all Internet traffic passes through the WCCP-configured router, and all intranet traffic is confined to 10.0.0.0/8. Because it is unlikely that remote Internet hosts have a Steelhead appliance, do not redirect Internet traffic to the Steelhead appliance. The following is an example ACL that achieves this goal.

Important: The following router command is not required for the example network configurations in this chapter. Use caution when you enter the **ip wccp [SG]** router command because each **ip wccp [SG]** router command overwrites the previous **ip wccp [SG]** router command. You cannot use an **ip wccp [SG]** router command to augment **ip wccp [SG]** router commands you previously issued. For details, see [“Configuring a Basic WCCP Router” on page 242](#).

To configure an ACL to route only intranet traffic to a WCCP-enabled Steelhead appliance interface

- On the WCCP router, at the system prompt, enter the following commands:

```
enable
configure terminal
access-list 101 deny ip host <WCCP_Steelhead_IP> any
access-list 101 deny ip any host <WCCP_Steelhead_IP>
access-list 101 permit tcp 10.0.0.0 0.255.255.255 any
access-list 101 permit tcp any 10.0.0.0 0.255.255.255
access-list 101 deny ip any any
!
ip wccp 61 redirect-list 101
!
end
write memory
```

Repeat these commands for each WCCP Steelhead appliance in the service group.

Note: Enter configuration commands, one per line. Enter CTRL-Z to end the configuration.

Configuring Load Balancing in WCCP

You can perform load balancing using WCCP. WCCP supports load balancing using either the hash assignment method or the mask assignment method. This section includes the following topics:

- [“Configuring Load Balancing Using the Hash Assignment Method” on page 249](#)
- [“Configuring Load Balancing Using the Mask Assignment Method” on page 249](#)
- [“Using the Weight Parameter” on page 251](#)

Configuring Load Balancing Using the Hash Assignment Method

With the hash assignment method, traffic is redirected based on a hashing scheme and the weight of the Steelhead appliance interfaces. You can hash on a combination of the source IP address, destination IP address, source port, or destination port. The default weight is based on the Steelhead appliance model (for example, for the Model 5000, the weight is 5000). You can modify the weight on an interface per service group.

To change the hashing scheme and assign a weight on a WCCP Steelhead appliance interface

1. Connect to the Riverbed CLI on the WCCP Steelhead appliance interface and enter the following command:

```
wccp interface inpath0_0 service-group 61 routers 10.1.0.1 flags dst-ip-hash,src-ip-hash
```

2. To change the weight on the WCCP Steelhead appliance interface, at the system prompt, enter the following command:

```
wccp interface inpath0_0 service-group 61 routers 10.1.0.1 weight 20
```

Configuring Load Balancing Using the Mask Assignment Method

Mask assignment uses 7 bits, which allows for a maximum of 128 buckets ($2^7 = 128$) for load balancing across Steelhead appliance interfaces in the same service group. When deciding the number of bits to use, always keep in mind the number of Steelhead appliance interfaces in the service group. Ensure that you create enough buckets for all the Steelhead appliance interfaces in the service group. For example, with three Steelhead appliances with two in-path interfaces each in a service group, use at least 3 bits for mask assignment to create 8 buckets ($2^3 = 8$). Having more buckets than Steelhead appliance interfaces is not a problem; in fact, it might be necessary to do so to distribute the load correctly. However, if there are more Steelhead appliance interfaces than available buckets, some Steelhead appliance interfaces remain idle.

Mask assignments have two subcategories:

- **Address mask** - A 4-byte value, each byte of which corresponds to each octet of the IP address.
- **Port mask** - A 2-byte value used to match the port number.

You can combine address masks with port masks, as long as the total number of bits used for the mask assignment value does not exceed 7 bits.

Note: The algorithm used for determining bucket allocation and assignment is vendor-specific; there is no common standard in the industry. The following explanation is specific to Steelhead appliances. Other vendors who support load distribution with mask assignment might use a different algorithm to distribute the loads amongst their own devices.

The default mask on the Steelhead appliance is 0x1741. Change this to suit your network. At a minimum, the number of bits you use in the mask must provide enough buckets to load balance the traffic among the Steelhead appliances in the cluster. In addition, make sure there are enough buckets created to fairly load balance the traffic.

When determining bucket allocations, mask assignment uses the WCCP *weight* parameter. The higher the weight, the more buckets are allocated to that Steelhead appliance interface. However, even if all the Steelhead appliance interfaces in the service group share the same weight, the distribution among the Steelhead appliance interfaces might not be perfectly equal if the number of buckets is not divisible by the number of Steelhead appliance interfaces in the service group.

A mask of 0x1 creates two buckets ($2^1=2$). This is appropriate for a deployment consisting of a single Steelhead appliance or a cluster of two. A mask of 0x3 creates four buckets ($2^2=4$), but is most likely not appropriate for a three Steelhead appliance deployment because it cannot lead to a fair distribution of the traffic.

When the number of buckets is not divisible by the number of Steelhead appliance interfaces in the service group, the remaining buckets are assigned to the Steelhead appliance interface with the highest IP address. In other words, the remainder from the following operation is assigned to the Steelhead appliance interface with the highest IP address:

$\langle \text{number of buckets} \rangle \text{ modulo } \langle \text{the number of Steelhead appliance interfaces} \rangle$

Effective weight with multiple in-path WCCP means each of the configured Steelhead appliance interface weights are divided by the number of that Steelhead appliance interfaces participating that service group. For example, a Steelhead appliance has two interfaces participating in service group 61. They have a weight of 100 configured. Their effective weight is $100/2$, or 50.

Example: Bucket Allocation for 8 Buckets and 3 Steelhead Appliance Interfaces of Equal Weight

When there are eight buckets and three Steelhead appliance interfaces of effective equal weight (Steelhead1 inpath0_0 (weight 200): 1.1.1.1, Steelhead1 inpath0_1 (weight 200): 2.2.2.2, and Steelhead2 inpath0_0 (weight 100): 3.3.3.3), the *initial* bucket allocation is:

- Interface 1.1.1.1 - two buckets
- Interface 2.2.2.2 - two buckets
- Interface 3.3.3.3 - two buckets

Using the expression $8 \bmod 3 = 2$, the remaining two buckets are assigned to Interface 3.3.3.3. The final allocation is:

- Interface 1.1.1.1 - two buckets (25%)
- Interface 2.2.2.2 - two buckets (25%)
- Interface 3.3.3.3 - four buckets (50%)

Example: Bucket Allocation for 16 Buckets and 3 Steelhead Appliance Interfaces of Equal Weight

The same operation applies to 16 buckets and 3 Steelhead appliance interfaces of equal effective weight.

Using the expression $16 \bmod 3 = 1$, the final allocation is:

- Interface 1.1.1.1 - five buckets (31.25%)
- Interface 2.2.2.2 - five buckets (31.25%)
- Interface 3.3.3.3 - six buckets (37.5%)

The example shows that the number of bits used for the mask and the number of Steelhead appliance interfaces in the service group affect the accuracy of the load distribution.

Using the Weight Parameter

To assign weight in the mask assignment method, you use the weight parameter in the same way as the hash assignment method: for example,

```
wccp interface inpath0_0 service-group 61 routers 10.1.0.1 weight 20
```

You can also assign weight to each Steelhead appliance interfaces so that the larger model Steelhead appliances are assigned more buckets. WCCP uses the following formula to assign buckets to each Steelhead appliance interface:

Bucket allocation = (bucket/size/sum of effective weight) * configured weight of the Steelhead appliance interface

Example: Bucket Allocation for 8 Buckets and 2 Steelhead Appliances with Single Interfaces with Different Weights

In this example, Steelhead A has a single in-path interface, inpath0_0, with a weight of 25 and Steelhead B has a single in-path interface, inpath0_0, with a weight of 50.

8 buckets

Total Steelhead appliance interface weight: $25 + 50 = 75$

Steelhead A inpath0_0 weight is 25

$(8/75) * 25 = 2.7$ buckets

Steelhead B inpath0_0 weight is 50

$(8/75) * 50 = 5.3$ buckets

However, because the number of allocated buckets must be integers, WCCP allocates two buckets to Steelhead A inpath0_0 and five buckets to Steelhead B inpath0_0. One unallocated bucket remains, so WCCP allocates it to the Steelhead appliance interface with the highest weight (Steelhead B inpath0_0), bringing the final bucket allocation for Steelhead B inpath0_0 to six buckets.

Example: Bucket Allocation for 16 Buckets and 3 different Steelhead Appliance Models, Each with Two In-Path Interfaces

In this example, Steelhead A has two interfaces, inpath0_0 with IP address of 10.1.1.1 and inpath0_1 with IP address 10.1.1.2, each configured with weight 25. Steelhead B has two interfaces, inpath0_0 with IP address of 10.1.1.3 and inpath0_1 with 10.1.1.4, each configured with weight 50. Steelhead C has inpath0_0 with IP address 10.1.1.5 and inpath0_1 with IP address 10.1.1.6, each configured with weight 75. 16 buckets.

The total weight equals the effective weight of all Steelhead appliance interfaces. The effective weight of a Steelhead appliance interface is equal to its configured weight divided by the number of that Steelhead appliance interfaces participating in the service group.

Total weight: $A\text{-in0_0}/2 + A\text{-in0_1}/2 + B\text{-in0_0}/2 + B\text{-in0_1} + C\text{-in0_0} + C\text{-in0_1}$

Total weight: $(25/2) + (25/2) + (50/2) + (50/2) + (75/2) + (75/2) = 150$

Initial Allocation:

Steelhead A inpath0_0 (10.1.1.1): $(12.5/150) * 16$ buckets = 1.33 = 1 bucket

Steelhead A inpath0_1 (10.1.1.2): $(12.5/150) * 16$ buckets = 1.33 = 1 bucket

Steelhead B inpath0_0 (10.1.1.3): $(25/150) * 16 \text{ buckets} = 2.66 = 2 \text{ buckets}$

Steelhead B inpath0_1 (10.1.1.4): $(25/150) * 16 \text{ buckets} = 2.66 = 2 \text{ buckets}$

Steelhead C inpath0_0 (10.1.1.5): $(37.5/150) * 16 \text{ buckets} = 4 = 4 \text{ buckets}$

Steelhead C inpath0_1 (10.1.1.6): $(37.5/150) * 16 \text{ buckets} = 4 = 4 \text{ buckets}$

Adding the initial bucket allocations, two buckets are still unassigned (16-14). These two remaining go to the highest IP address, Steelhead C inpath0_1, totaling six buckets.

Flow Data in WCCP

In virtual in-path deployments such as WCCP, traffic moves in and out of the same WAN interface. The LAN interface is not used. When the Steelhead appliance exports data to a data flow collector, all traffic has the WAN interface index. Although it is technically correct for all traffic to have the WAN interface index because the input and output interfaces are the same, it is impossible to use the interface index to distinguish between LAN-to-WAN and WAN-to-LAN traffic.

You can configure the fake index feature on your Steelhead appliance to insert the correct interface index before exporting data to a data flow collector.

For details on configuring the fake index feature, see [“Configuring Flow Data Exports in Virtual In-Path Deployments” on page 220](#).

Verifying and Troubleshooting WCCP Configurations

This section describes the basic commands for verifying WCCP configuration on the router and the WCCP Steelhead appliance.

To verify the router configuration

- On the router, at the system prompt, enter the following commands:

```
enable
show ip wccp
show ip wccp 61 detail
show ip wccp 61 view
```

To verify the WCCP configuration on an interface

- On the router, at the system prompt, enter the following commands:

```
enable
show ip interface
```

Look for WCCP status messages near the end of the output.

To verify the access control list configuration

- On the router, at the system prompt, enter the following commands:

```
enable
show access-lists <access_list_number>
```

To trace WCCP packets and events on the router

- On the router, at the system prompt, enter the following commands:

```
enable
debug ip wccp events
WCCP events debugging is on
debug ip wccp packets
WCCP packet info debugging is on
term mon
```

To verify the WCCP Steelhead appliance configuration

- Connect to the Riverbed CLI on the WCCP Steelhead appliance and enter the following command:

```
show wccp interface <interface> service-group 61 detail
WCCP Support Enabled:          yes
WCCP Multicast TTL:            1
WCCP Return via Gateway Override: no

Router IP Address:              89.1.1.1
Identity:                      1.1.1.1
State:                         Connected
Redirect Negotiated:           12
Return Negotiated:             12
Assignment Negotiated:         mask
i-see-you Message Count:       20
Last i-see-you Message:        2008/07/06 22:05:16 (1 second(s) ago)
Removal Query Message Count:   0
Last Removal Query Message:    N/A (0 second(s) ago)
here-i-am Message Count:       20
Last here-i-am Message:        2008/07/06 22:05:16 (1 second(s) ago)
Redirect Assign Message Count:  1
Last Redirect Assign Message:   2008/07/06 22:02:21 (176 second(s) ago)

Web Cache Client Id: 89.1.1.2
Weight:                  25
Distribution:             1 (25.00%)

Mask      SrcAddr      DstAddr      SrcPort      DstPort
----      -
0000:    0x02000000    0x00000000    0x0000    0x0001

Value      SrcAddr      DstAddr      SrcPort      DstPort      Cache-IP
----      -
0000:    0x00000000    0x00000000    0x0000    0x0000    89.1.1.2

Web Cache Client Id: 89.1.1.6
Weight:                  25
Distribution:             2 (50.00%)

Mask      SrcAddr      DstAddr      SrcPort      DstPort
----      -
0000:    0x02000000    0x00000000    0x0000    0x0001

Value      SrcAddr      DstAddr      SrcPort      DstPort      Cache-IP
----      -
0002:    0x00000000    0x00000000    0x0000    0x0001    89.1.1.6
0003:    0x02000000    0x00000000    0x0000    0x0001    89.1.1.6

Web Cache Client Id: 89.1.1.5
Weight:                  25
Distribution:             1 (25.00%)

Mask      SrcAddr      DstAddr      SrcPort      DstPort
```

```

-----
0000:  0x02000000  0x00000000  0x0000  0x0001

Value  SrcAddr      DstAddr      SrcPort  DstPort  Cache-IP
-----
0001:  0x02000000  0x00000000  0x0000  0x0000  89.1.1.5

```

To verify the WCCP bucket allocation

- Connect to the Riverbed CLI on the WCCP Steelhead appliance and enter **show wccp interface <interface> service-group 61 detail**.

This command is available only in RiOS v6.1 or later.

The following is an example of WCCP bucket allocation details.

- WCCP used 2 bits for the mask, 1 in the Source Address and 1 in the Destination Port:

```

Mask      SrcAddr      DstAddr      SrcPort  DstPort
-----
0000:  0x02000000  0x00000000  0x0000  0x0001

```

- WCCP created four buckets ($2^2 = 4$) and allocated them to the Steelhead appliance interfaces as follows:

89.1.1.2 was allocated one bucket:

```

Value  SrcAddr      DstAddr      SrcPort  DstPort  Cache-IP
-----
0000:  0x00000000  0x00000000  0x0000  0x0000  89.1.1.2

```

89.1.1.5 was allocated one bucket:

```

Value  SrcAddr      DstAddr      SrcPort  DstPort  Cache-IP
-----
0001:  0x02000000  0x00000000  0x0000  0x0000  89.1.1.5

```

89.1.1.6 was allocated two buckets because it has the highest IP address of the attached Steelhead appliance interfaces:

```

Value  SrcAddr      DstAddr      SrcPort  DstPort  Cache-IP
-----
0002:  0x00000000  0x00000000  0x0000  0x0001  89.1.1.6
0003:  0x02000000  0x00000000  0x0000  0x0001  89.1.1.6

```

The following table lists some of the configurations that the **show wccp service-group <num> details** CLI command displays.

Configuration	Example
Redirection Method	Redirect Negotiated: 12
Return Method	Return Negotiated: 12
Assignment Method	Assignment Negotiated: mask
GRE Encapsulation	WCCP Return via Gateway Override: no
WCCP Control Messages	i-see-you Message Count: 20

For details on troubleshooting WCCP and other deployments, see [“Troubleshooting Steelhead Appliance Deployment Problems” on page 431](#).

CHAPTER 10 Policy-Based Routing Virtual In-Path Deployments

This chapter describes the basic steps for PBR network deployments and how to configure policy-based routing (PBR) to redirect traffic to a Steelhead appliance or group of Steelhead appliances.

This chapter includes the following sections:

- [“Overview of PBR” on page 255](#)
- [“Connecting the Steelhead Appliance in a PBR Deployment” on page 258](#)
- [“Configuring PBR” on page 258](#)
- [“Exporting Flow Data and Virtual In-Path Deployments” on page 273](#)

For details on the factors you must consider before you design and deploy the Steelhead appliance in a network environment, see [“Choosing the Right Steelhead Appliance” on page 16](#).

Overview of PBR

PBR is a packet redirection mechanism that allows you to define policies to route packets instead of relying on routing protocols. PBR redirects packets to Steelhead appliances that are in a virtual in-path deployment. This section includes the following topics:

- [“PBR Failover and Cisco Discovery Protocol” on page 256](#)
- [“Alternate PBR Failover Mechanisms” on page 257](#)

You define PBR policies on your router for switching packets. PBR policies can be based on identifiers available in access lists, such as the source IP address, destination IP address, protocol, source port, or destination port.

When a PBR-enabled router interface receives a packet that matches a defined policy, PBR switches the packet according to the rule defined for the policy. If a packet does not match a defined policy, the packet is routed by the IP address specified in the routing table entry that most closely matches the destination address.

Important: To avoid an infinite loop, PBR must be enabled on the router interfaces where client traffic arrives, and disabled on the router interface that is connected to the Steelhead appliance.

PBR is enabled as a global configuration and applied on an interface basis. Each virtual in-path interface can be used simultaneously for receiving traffic redirected through PBR; physically, the WAN port is cabled and used to receive the redirected traffic.

The Steelhead appliance that intercepts traffic redirected by PBR is configured with both in-path and virtual in-path support enabled.

PBR Failover and Cisco Discovery Protocol

A major issue with PBR is that it can cause a traffic black hole; that is, it drops all packets to a destination if the device it is redirecting to fails. You can avoid the traffic black holes by enabling PBR to track whether or not the PBR next-hop IP address is available. You configure the PBR-enabled router to use the Cisco Discovery Protocol (CDP). CDP is a protocol used by Cisco routers and switches to obtain information such as neighbor IP addresses, models, and IOS versions. The protocol runs at the OSI Layer-2 using the 802.3 Ethernet frame. You also enable CDP on the Steelhead appliance.

CDP must be enabled on the Steelhead appliance that is used in the PBR deployment. You enable CDP using the **in-path cdp enable** CLI command. For details, see the *Riverbed Command-Line Interface Reference Manual*.

CDP enables Steelhead appliances to provide automatic failover for PBR deployments. You configure the Steelhead appliance to send out CDP frames. The PBR-enabled router uses these frames to determine whether the Steelhead appliance is operational. If the Steelhead appliance is not operational, the PBR-enabled router stops receiving the CDP frames, and PBR stops switching traffic to the Steelhead appliance.

The Steelhead appliance must be physically connected to the PBR-enabled router for CDP to send frames. If a switch or other Layer-2 device is located between the PBR-enabled router and the Steelhead appliance, CDP frames cannot reach the router. If the CDP frames do not reach the router, the router assumes that the Steelhead appliance is not operational.

CDP is not supported as a failover mechanism on all Cisco platforms. For details on whether your Cisco device supports this feature, refer to your router documentation.

To enable CDP on the Steelhead appliance

- Connect to the Steelhead CLI and enter the following commands:

```
enable
configure terminal
in-path cdp enable
write memory
restart
```

Note: You must save your changes and restart the Steelhead appliance for your changes to take effect.

To enable CDP failover on the router

- On the PBR router, at the system prompt, use the **set ip next-hop verify-availability** command. For details, refer to your router documentation.

ICMP and HTTP GET can both also be used to track whether or not the PBR next-hop IP address is available.

When you configure the **set ip next-hop verify-availability** Cisco router command, PBR sends a packet in the following manner:

- PBR checks the CDP neighbor table to verify that the PBR next-hop IP address is available.

- If the PBR next-hop IP address is available, PBR sends an ARP request for the address, obtains an answer for it, and redirects traffic to the PBR next-hop IP address (the Steelhead appliance).
- PBR continues sending traffic to the next-hop IP address as long as the ARP requests obtain answers for the next-hop IP address.
- If the ARP request fails to obtain an answer, PBR checks the CDP table. If there is no entry in the CDP table, PBR stops using the route map to send traffic. This verification provides a failover mechanism.

A Cisco 6500 router and switch combination that is configured in hybrid mode does not support PBR with CDP. A hybrid setup requires that you use a native setup for PBR with CDP to work. This configuration fails because all routing is performed on the MSFC. The MSFC is treated as an independent system in a hybrid setup. Therefore, when you run the **show cdp neighbors** Cisco command on the MSFC, it displays the supervisor card as its only neighbor. PBR does not detect the devices that are connected to the switch ports. As a result, PBR does not redirect any traffic for route maps that use the **set ip next-hop verify-availability** Cisco command. For details, refer to your router documentation.

Alternate PBR Failover Mechanisms

Several other PBR failover methods exist:

- Object tracking
- Dedicated layer-3 subnet
- Scriptable programming language

Object tracking is a Cisco IOS feature. The Cisco router generates synthetic traffic through a variety of methods (HTTP GET, ping, TCP connect, and so on) to determine if the Steelhead appliance interface is available. If the Steelhead appliance interface is declared unavailable by object tracking, then the router moves to the next Steelhead appliance, or routes the packet normally.

For more information about object tracking, see [“Configuring a Steelhead Appliance with Object Tracking” on page 263](#).

You can deploy the Steelhead appliance on a dedicated Layer-3 subnet. A dedicated Layer-3 subnet provides a simple approach to failover without incurring additional CPU load on the Layer-3 device. The Steelhead appliance must be the only device in the subnet, and connected to the Layer-3 device. If the Steelhead appliance becomes unavailable, the dedicated Layer-3 subnet disappears from the routing table and the packet is routed normally. When all the interfaces for a VLAN or subnet do not connect to a Layer-3 switch, the corresponding route is withdrawn from the routing table and the policy statement is bypassed.

If you use a Layer-3 switch, there cannot be another interface in the WAN optimization VLAN, including 802.1Q trunks.

Some Layer-3 devices include a scriptable programming language: for example, Cisco Embedded Event Manager. You can use a scriptable programming language to actively detect an event, and initiate a series of CLI commands to disable PBR. If an event occurs indicating the Steelhead appliance has failed, the PBR configuration is automatically removed. If the event reverses, the PBR configuration is automatically re-applied.

Connecting the Steelhead Appliance in a PBR Deployment

You can use several types of Ethernet cables to attach to the Steelhead appliance in PBR deployments:

- A straight-through cable to the primary interface. You use this connection to manage the Steelhead appliance, reaching it through HTTPS or SSH.
- A straight-through cable to the WAN0_0 interface if you are connecting to a switch.
- A crossover cable to the WAN0_0 interface if you are connecting to a router. You assign an IP address to the in-path interface; this is the IP address that you redirect traffic to (the target of the router PBR rule).

Configuring PBR

This section describes how to configure PBR and provides example deployments. This section includes the following topics:

- [“Overview of Configuring PBR” on page 258](#)
- [“Configuring a Steelhead Appliance to Directly Connect to the Router” on page 259](#)
- [“Configuring a Steelhead Appliance to Connect to Layer-2 Switch” on page 260](#)
- [“Configuring a Steelhead Appliance to Connect to a Layer-3 Switch” on page 262](#)
- [“Configuring a Steelhead Appliance with Object Tracking” on page 263](#)
- [“Configuring a Steelhead Appliance with Multiple PBR Interfaces” on page 264](#)
- [“Configuring Multiple Steelhead Appliances to Connect to Multiple Routers” on page 265](#)
- [“Configuring PBR for Load-Balancing WAN Circuits” on page 268](#)
- [“Configuring Local PBR for ICMP Redirection in a Mixed MTU Environment” on page 272](#)

Overview of Configuring PBR

You can use access lists to specify which traffic is redirected to the Steelhead appliance. Traffic that is not specified in the access list is switched normally. If you do not have an access list, or if your access list is not correctly configured in the route map, traffic is not redirected.

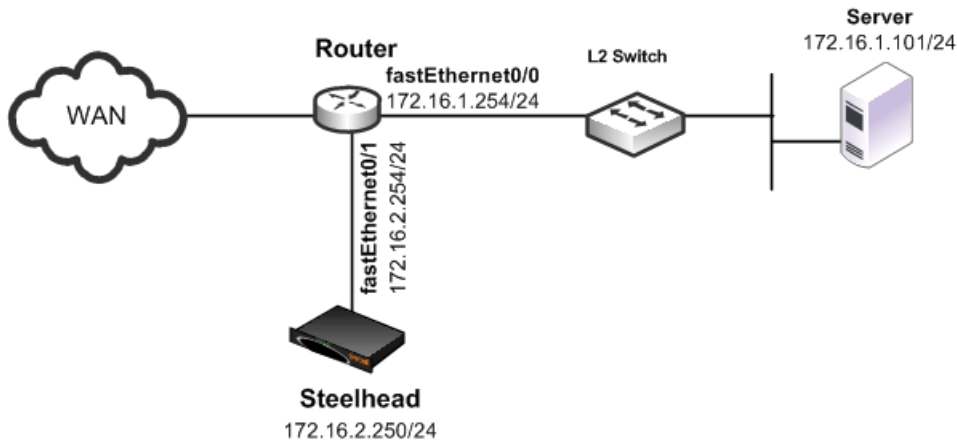
For details on access lists, see [“Configuring Access Control Lists” on page 246](#).

Important: Riverbed recommends that you define a policy based on the source or destination IP address rather than on the TCP source or destination ports, because certain protocols use dynamic ports instead of fixed ones.

Configuring a Steelhead Appliance to Directly Connect to the Router

Figure 10-1 shows a Steelhead appliance deployment in which the Steelhead appliance is configured with PBR and is directly connected to the router.

Figure 10-1. Steelhead Appliance Directly Connected to the Router



In this example:

- The router fastEthernet0/0 interface is attached to the Layer-2 switch.
- The router fastEthernet0/1 interface is attached to the Steelhead appliance.
- A single Steelhead appliance is configured. You can add more Steelhead appliances using the same method as for the first Steelhead appliance.

Although the primary interface is not included in this example, Riverbed recommends, as a best practice, that you connect the primary interface for management purposes.

You must configure subnet side rules when you use RSP or VSP in a virtual in-path deployment. To configure subnet side rules, select **Configure > Networking > Subnet Side**. VSP is enabled by default in the Steelhead EX series.

For details on configuring the primary interface, see the *Steelhead Appliance Management Console User's Guide*.

To configure a Steelhead appliance with PBR connected directly to the router

1. Connect to the Steelhead CLI and enter the following commands:

```

enable
configure terminal
in-path enable
in-path oop enable
interface in-path0_0 ip address 172.16.2.250/24
ip in-path-gateway inpath0_0 172.16.2.254
write memory
restart
  
```

You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

2. On the PBR router, at the system prompt, enter the following commands:

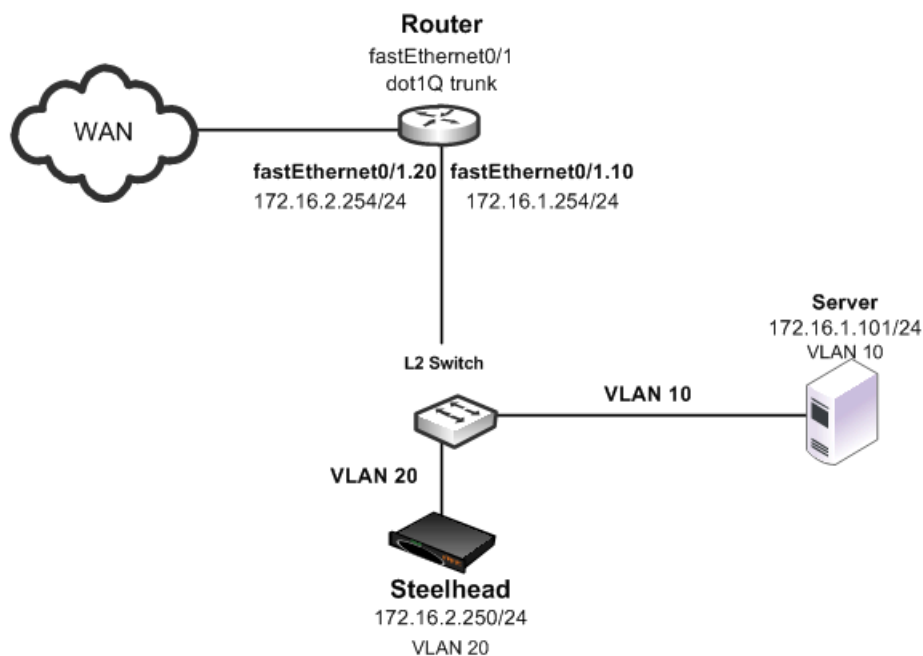
```
enable
configure terminal
route-map riverbed
match ip address 101
set ip next-hop 172.16.2.250
exit
ip access-list extended 101
permit tcp any 172.16.1.101 0.0.0.0
permit tcp 172.16.1.101 0.0.0.0 any
exit
interface fa0/0
ip policy route-map riverbed
interface S0/0
ip policy route-map riverbed
exit
exit
write memory
```

Enter one configuration command per line. Press CTRL-Z to end the configuration.

Configuring a Steelhead Appliance to Connect to Layer-2 Switch

Figure 10-2 shows a Steelhead appliance deployment in which the Steelhead appliance is configured with PBR and is directly connected to the router through a switch. This deployment also has a trunk between the switch and the router.

Figure 10-2. Steelhead Appliance Connected to a Layer-2 Switch with a VLAN



In this example:

- The switch logically separates the server and the Steelhead appliance by placing:
 - the server on VLAN 10.
 - the Steelhead appliance on VLAN 20.
- The router fastEthernet0/1 interface is attached to the Layer-2 switch.
- The router performs inter-VLAN routing; that is, the router switches packets from one VLAN to the other.
- The link between the router and the switch is configured as a **dot1Q** trunk to transport traffic from multiple VLANs.

Although the primary interface is not included in this example, Riverbed recommends that you connect the primary interface for management purposes.

For details on configuring the primary interface, see the *Steelhead Appliance Management Console User's Guide*.

To configure a Steelhead appliance with PBR to a connected Layer-2 switch with a VLAN to the router

1. Connect to the Steelhead CLI and enter the following commands:

```
enable
configure terminal
in-path enable
in-path oop enable
interface in-path0_0 ip address 172.16.2.250/24
ip in-path-gateway inpath0_0 172.16.2.254
write memory
restart
```

You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

2. On the PBR router, at the system prompt, enter the following commands:

```
enable
configure terminal
route-map riverbed
match ip address 101
set ip next-hop 172.16.2.250
exit
ip access-list extended 101
permit tcp any 172.16.1.101 0.0.0.0
permit tcp 172.16.1.101 0.0.0.0 any
exit
interface fa0/1.10
encapsulation dot1Q 10
ip address 172.16.1.254 255.255.255.0
interface fa0/1.20
encapsulation dot1Q 20
ip address 172.16.2.254 255.255.255.0
exit
interface fa0/1.10
ip policy route-map riverbed
interface S0/0
ip policy route-map riverbed
```

```

exit
exit
write memory

```

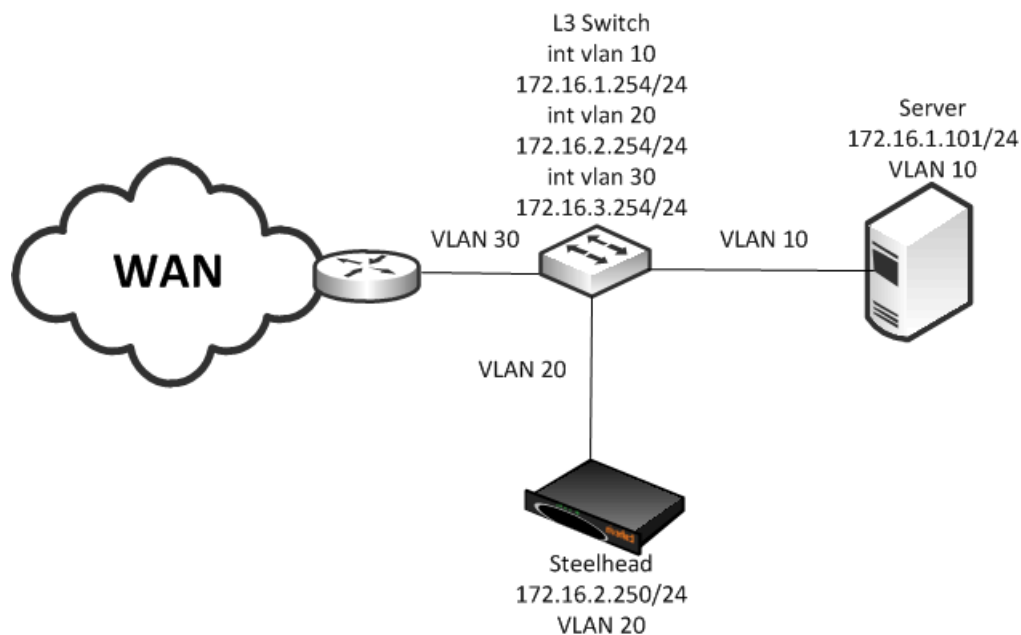
Tip: Enter one configuration command per line. Press CTRL-Z to end the configuration.

Note: In this example, assume that both the Steelhead appliance and the server are connected to the correct VLAN. Also assume that these VLAN connections are established through the switch port configuration on the Layer-2 switch.

Configuring a Steelhead Appliance to Connect to a Layer-3 Switch

Figure 10-3 shows a Steelhead appliance deployment in which the Steelhead appliance is configured with PBR and is directly connected to a Layer-3 switch.

Figure 10-3. Steelhead Appliance Connected to a Layer-3 Switch



In this example:

- The Layer-3 switch fastEthernet0/0 interface is attached to the server and is on VLAN 10.
- The Layer-3 switch fastEthernet0/1 interface is attached to the Steelhead appliance and is on VLAN 20.
- A single Steelhead appliance is configured. You can add more appliances using the same method as used for the first Steelhead appliance.

Note: Although the primary interface is not included in this example, Riverbed recommends that you connect the primary interface for management purposes. For details on configuring the primary interface, see the *Steelhead Appliance Management Console User's Guide*.

To configure a Steelhead appliance with PBR connected directly to a Layer-3 switch

1. Connect to the Steelhead CLI and enter the following commands:

```
enable
configure terminal
in-path enable
in-path oop enable
in-path cdp enable
interface in-path0_0 ip address 172.16.2.250/24
ip in-path-gateway inpath0_0 172.16.2.254
write memory
restart
```

You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

2. On the Layer-3 switch, at the system prompt, enter the following commands:

```
enable
configure terminal
route-map riverbed
match ip address 101
set ip next-hop 172.16.2.250
set ip next-hop verify-availability
exit
ip access-list extended 101
permit tcp any 172.16.1.101 0.0.0.0
permit tcp 172.16.1.101 0.0.0.0 any
exit
interface vlan 10
ip address 172.16.1.254 255.255.255.0
ip policy route-map riverbed
interface vlan 20
ip address 172.16.2.254 255.255.255.0
interface vlan 30
ip policy route-map riverbed
exit
exit
write memory
```

Tip: Enter one configuration commands per line. Press CTRL-Z to end the configuration.

Configuring a Steelhead Appliance with Object Tracking

In an object tracking deployment, the Steelhead appliance is connected to the router, and the router tracks whether the Steelhead appliance is reachable using the Object Tracking feature of Cisco IOS. Object Tracking enables you to use methods such as HTTP GET and ping, to determine whether the PBR next-hop IP address is available.

Object Tracking is not available on all Cisco devices. For details on whether your Cisco device supports this feature, refer to your router documentation.

To configure the Steelhead appliance with object tracking

For diagram details, see [Figure 10-1](#).

1. Connect to the Steelhead CLI and enter the following commands:

```
enable
configure terminal
interface in-path0_0 ip address 172.16.2.2
50/24
ip in-path-gateway inpath0_0 172.16.2.254
in-path enable
in-path oop enable

no interface inpath0_0 fail-to-bypass enable
write memory
```

2. On the PBR router, at the system prompt, enter the following commands:

```
enable
configure terminal
ip sla 1
icmp-echo 172.16.2.250
ip sla schedule 1 life forever start-time now
track 101 rtr 1 reachability
route-map riverbed
match ip address 101
set ip next-hop verify-availability 172.16.2.250 10 track 101
exit
ip access-list extended 101
permit tcp any 172.16.1.101 0.0.0.0
permit tcp 172.16.1.101 0.0.0.0 any
exit
interface fa0/0
ip policy route-map riverbed
interface S0/0
ip policy route-map riverbed
exit
exit
write memory
```

Configuring a Steelhead Appliance with Multiple PBR Interfaces

In a deployment that uses multiple PBR interfaces, the Steelhead appliance is connected to two routers, each of which is configured to redirect traffic to a separate interface on the Steelhead appliance. Each router is configured similarly to the single router deployment, except that you specify a next-hop IP address that corresponds to the interface to which the Steelhead appliance connects.

To configure the Steelhead appliance with multiple PBR interfaces

For diagram details, see [Figure 10-1](#).

1. Connect to the Steelhead CLI and enter the following commands:

```
enable
configure terminal
in-path enable
Interface inpath0_1 enable
in-path oop enable
interface in-path0_0 ip address 172.16.2.250/24
ip in-path-gateway inpath0_0 172.16.2.254
interface in-path0_1 ip address 172.16.3.250/24
```

```
ip in-path-gateway inpath0_1 172.16.3.254
write memory
restart
```

You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

2. On the first PBR router, at the system prompt, enter the following commands:

```
enable
configure terminal
route-map riverbed
match ip address 101
set ip next-hop 172.16.2.250
exit
ip access-list extended 101
permit tcp any 172.16.1.101 0.0.0.0
permit tcp 172.16.1.101 0.0.0.0 any
exit
interface fa0/0
ip policy route-map riverbed
interface S0/0
ip policy route-map riverbed
exit
exit
write memory
```

3. On the second PBR router, at the system prompt, enter the following commands:

```
enable
configure terminal
route-map riverbed
match ip address 101
set ip next-hop 172.16.3.250
exit
ip access-list extended 101
permit tcp any 172.16.1.101 0.0.0.0
permit tcp 172.16.1.101 0.0.0.0 any
exit
interface fa0/0
ip policy route-map riverbed
interface S0/0
ip policy route-map riverbed
```

Configuring Multiple Steelhead Appliances to Connect to Multiple Routers

In a PBR environment, you can deploy multiple Steelhead appliances for optimization redundancy. [Figure 10-4](#) shows a Steelhead appliance deployment in which two routers are redirecting packets to two Steelhead appliances. The Steelhead appliances are directly connected through multiple interfaces. This deployment provides a high-availability environment—a full-mesh topology between the routers and Steelhead appliances.

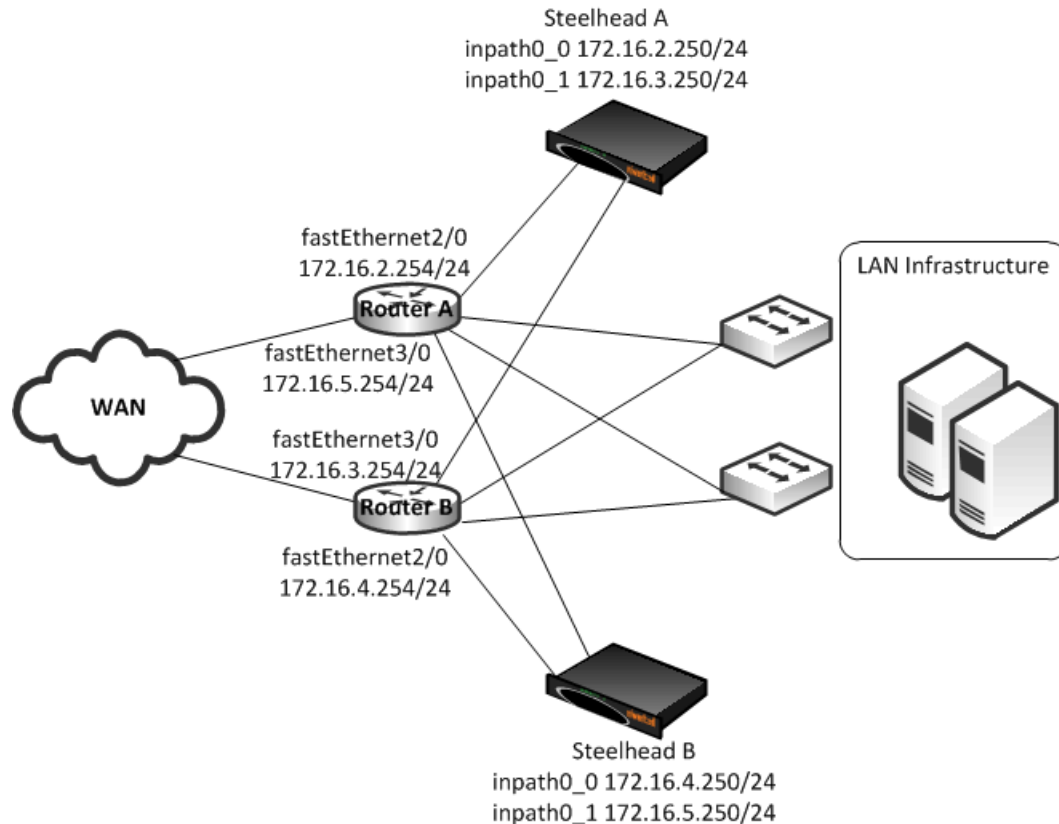
When you deploy multiple Steelhead appliances, the routers can redirect packets within a TCP connection to different Steelhead appliances, depending on the router policy. For example, Router A redirects packets to Steelhead A. Following network convergence due to a WAN outage packets for this TCP connection, the packets arrive on Router B. Router B has a policy configured to redirect packets to Steelhead B. In this type of environment, Riverbed recommends that you use connection forwarding to ensure packets within a flow are forwarded to the owning Steelhead appliance for optimization.

For more details on connection forwarding, see [“Connection Forwarding” on page 42](#).

Data store synchronization is not a requirement for Steelhead appliances in a PBR environment, but it is useful if your design goal is for warm performance after a Steelhead appliance fails. To use data store synchronization, you must meet certain criteria.

For more details on the data store, see [“RiOS Data Store Synchronization”](#) on page 15.

Figure 10-4. Multiple Steelhead Appliances Directly Connected to Dual Routers



The example in [Figure 10-4](#) shows the following:

- Each router fastEthernet2/0 interface is attached to the Steelhead appliance wan0_0 interface using the inpath0_0 IP address.
- Each router fastEthernet3/0 interface is attached to the Steelhead appliance wan0_1 interface using the inpath0_1 IP address.
- Each router withdraws the PBR next-hop statement because the Steelhead appliances are directly connected (in case a Steelhead appliance interface or appliance fail). You can use other methods for failover, such as CDP, object tracking, or embedded event manager.
- Connection forwarding is enabled to ensure the owning Steelhead appliance receives all packets for the TCP connection.

You can redirect packets to Steelhead appliance interfaces in any order. [Figure 10-4](#) shows Router A redirecting packets first to Steelhead A inpath0_0, and then to Steelhead B inpath0_1; and Router B redirecting packets first to Steelhead B inpath0_0 then Steelhead A inpath0_1. Router A and Router B can redirect to Steelhead A inpath0_0 and inpath0_1 respectively.

This example shows two Steelhead appliances. You can add more Steelhead appliances using a similar method as for the first Steelhead appliance.

To configure dual Steelhead appliances with PBR connected to dual routers

1. Connect to Steelhead A CLI and enter the following commands:

```
enable
configure terminal
in-path enable
in-path oop enable
interface in-path0_0 ip address 172.16.2.250/24
ip in-path-gateway inpath0_0 172.16.2.254
interface in-path0_1 ip address 172.16.3.250/24
ip in-path-gateway inpath0_1 172.16.3.254
in-path neighbor enable
in-path neighbor multi-interface enable
in-path name SteelheadB main-ip 172.16.3.250
in-path name SteelheadB additional-ip 172.16.4.250
in-path neighbor allow-failure
no interface inpath0_0 fail-to-bypass enable
no interface inpath0_1 fail-to-bypass enable
write memory
restart
```

You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

2. Connect to Steelhead B CLI and enter the following commands:

```
enable
configure terminal
in-path enable
in-path oop enable
interface in-path0_0 ip address 172.16.2.250/24
ip in-path-gateway inpath0_0 172.16.2.254
interface in-path0_1 ip address 172.16.3.250/24
ip in-path-gateway inpath0_1 172.16.3.254
in-path neighbor enable
in-path neighbor multi-interface enable
in-path name SteelheadA main-ip 172.16.2.250
in-path name SteelheadA additional-ip 172.16.5.250
in-path neighbor allow-failure
no interface inpath0_0 fail-to-bypass enable
no interface inpath0_1 fail-to-bypass enable
write memory
restart
```

You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

3. On Router A, at the system prompt, enter the following commands:

```
enable
configure terminal
route-map riverbed
match ip address 101
set ip next-hop 172.16.2.250 172.16.5.250
exit
ip access-list extended 101
permit tcp any 172.16.1.101 0.0.0.0
permit tcp 172.16.1.101 0.0.0.0 any
exit
interface fa2/0
ip address 172.16.2.254 255.255.255.0
interface fa3/0
ip address 172.16.5.254 255.255.255.0
```

```

exit
interface fa1/0
ip policy route-map riverbed
interface S0/0
ip policy route-map riverbed
exit
exit
write memory

```

4. On Router B, at the system prompt, enter the following commands:

```

enable
configure terminal
route-map riverbed
match ip address 101
set ip next-hop 172.16.4.250 172.16.3.250
exit
ip access-list extended 101
permit tcp any 172.16.1.101 0.0.0.0
permit tcp 172.16.1.101 0.0.0.0 any
exit
interface fa2/0
ip address 172.16.3.254 255.255.255.0
interface fa3/0
ip address 172.16.4.254 255.255.255.0
exit
interface fa1/0
ip policy route-map riverbed
interface S0/0
ip policy route-map riverbed
exit
exit
write memory

```

Configuring PBR for Load-Balancing WAN Circuits

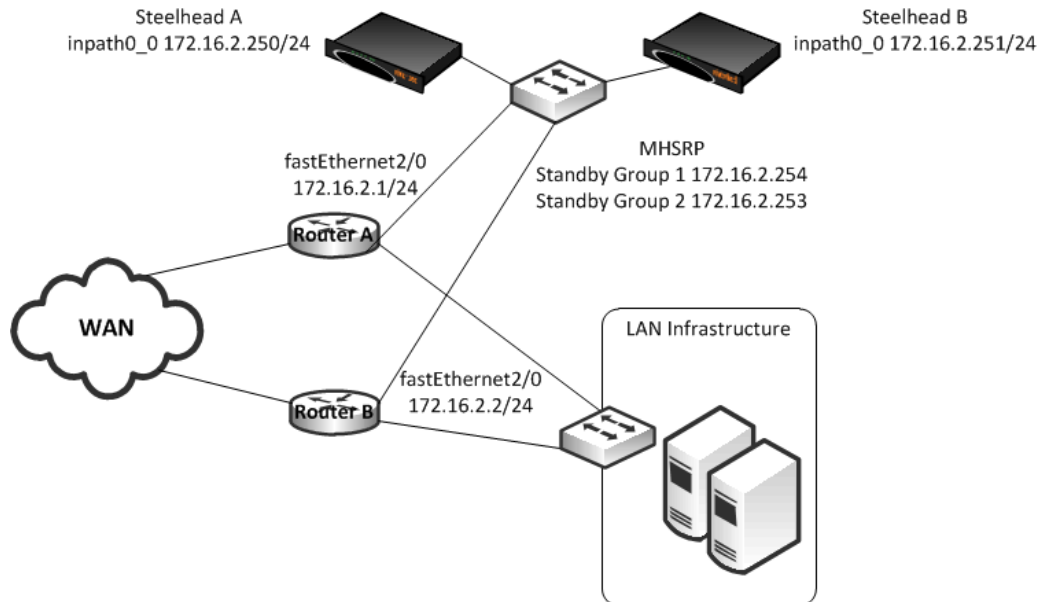
In a network with multiple entry and exit points, you can configure the router and Steelhead appliance to support two goals:

- Support peer affinity so one Steelhead appliance more efficiently optimizes the traffic for a remote peer
- Maintain outbound load balancing by allowing the Steelhead appliances to use separate WAN circuits to reach the remote sites

Figure 10-5 shows all Steelhead appliances in one subnet. The routers are configured with access control lists (ACLs) to direct traffic from remote peer A to Steelhead A, and traffic for remote peer B to Steelhead B. The failover method is object tracking. An optional all-inclusive rule is added to simplify the addition of a new remote site. To support outbound load balancing, each Steelhead appliance uses a different default gateway.

Optimized traffic from Steelhead A reaches the WAN through Router A, and Steelhead B reaches the WAN through router B. Multiple HSRP (MHSRP) allows the routers to present two default gateways for the WAN optimization subnet, at the same time having redundancy in case a WAN circuit fails. Instead of MHSRP, you could use Gateway Load Balancing Protocol (GLBP) or static routes on the Steelhead appliances. If you use one of these two methods, the Steelhead appliances are unaware of a WAN circuit outage.

Figure 10-5. PBR Load-Balancing



In this example:

- Each router fastEthernet2/0 interface and Steelhead wan0_0 interface are attached to same VLAN on the switch.
- Each router uses object tracking to bypass the PBR next-hop statement because the Steelhead appliances are not directly connected to the router. Embedded event manager is another option.
CDP frames from the Steelhead appliance are consumed by the switch and do not reach the routers. This renders CDP ineffective for failover.
- Packets are redirected to a Steelhead appliance based on remote IP address subnet for peer affinity.
- Multiple HSRP groups are used on the router fastEthernet2/0 interfaces to achieve outbound load balancing on the WAN. (Gateway load balancing protocol, or configuring static routes on the Steelhead appliance in-path interface can be used instead of multiple HSRP.)
- Router WAN interfaces are tracked when using HSRP to ensure the router has a path to the remote site.
- Multi-interface is enabled in case another Steelhead appliance interface is connected to another WAN optimization subnet.

Two Steelhead appliances are configured in this example. You can add more Steelhead appliances using the same method as used for one of the Steelhead appliances ensuring policy route statements, any ACLs, default gateway align with load-balancing and redundancy goals.

To configure PBR appliances in a load-balanced environment

1. Connect to Steelhead A CLI and enter the following commands:

```
enable
configure terminal
```

```

in-path enable
in-path oop enable
interface in-path0_0 ip address 172.16.2.250/24
ip in-path-gateway inpath0_0 172.16.2.254
in-path neighbor enable
in-path neighbor multi-interface enable
in-path name SteelheadB main-ip 172.16.2.251
in-path neighbor allow-failure
no interface inpath0_0 fail-to-bypass enable
write memory
restart

```

You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

2. Connect to Steelhead B CLI and enter the following commands:

```

enable
configure terminal
in-path enable
in-path oop enable
interface in-path0_0 ip address 172.16.2.251/24
ip in-path-gateway inpath0_0 172.16.2.253
in-path neighbor enable
in-path neighbor multi-interface enable
in-path name SteelheadA main-ip 172.16.2.250
in-path neighbor allow-failure
no interface inpath0_0 fail-to-bypass enable
write memory
restart

```

You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

3. On PBR router A, at the system prompt, enter the following commands:

```

enable
configure terminal
ip access-list extended Remotes-To-A
permit tcp address 10.1.100.0 0.0.0.255 any
permit tcp address any 10.1.100.0 0.0.0.255
exit
ip access-list extended Remotes-To-B
permit tcp address 10.1.200.0 0.0.0.255 any
permit tcp address any 10.1.200.0 0.0.0.255
exit
ip access-list extended Catch-All
permit tcp any 172.16.1.101 0.0.0.0
permit tcp 172.16.1.101 0.0.0.0 any
exit
ip sla 1
icmp-echo 172.16.2.250 source-interface fastEthernet2/0
exit
ip sla schedule 1 life forever start-time now
ip sla 2
icmp-echo 172.16.2.251 source-interface fastEthernet2/0
exit
ip sla schedule 2 life forever start-time now
track 1 ip sla 1 reachability
track 2 ip sla 2 reachability
track 100 interface S0/0 line-protocol
route-map riverbed permit 10
match ip address Remotes-To-A
set ip next-hop verify-availability 172.16.2.250 1 track 1

```



```

set ip next-hop verify-availability 172.16.2.251 2 track 2
route-map riverbed permit 20
match ip address Remotes-To-B
set ip next-hop verify-availability 172.16.2.251 1 track 2
set ip next-hop verify-availability 172.16.2.250 2 track 1
route-map riverbed permit 30
match ip address Catch-All
set ip next-hop verify-availability 172.16.2.250 1 track 1
set ip next-hop verify-availability 172.16.2.251 2 track 2
exit
interface fa2/0
ip address 172.16.2.1 255.255.255.0
standby 1 ip 172.16.2.254
standby 1 priority 110
standby 1 preempt
standby 1 track 100 20
standby 2 ip 172.16.2.253
standby 2 preempt
exit
interface fa1/0
ip policy route-map riverbed
interface S0/0
ip policy route-map riverbed
exit
exit
write memory

```

4. On PBR Router B, at the system prompt, enter the following commands:

```

enable
configure terminal
ip access-list extended Remotes-To-A
permit tcp address 10.1.100.0 0.0.0.255 any
permit tcp address any 10.1.100.0 0.0.0.255
exit
ip access-list extended Remotes-To-B
permit tcp address 10.1.200.0 0.0.0.255 any
permit tcp address any 10.1.200.0 0.0.0.255
exit
ip access-list extended Catch-All
permit tcp any 172.16.1.101 0.0.0.0
permit tcp 172.16.1.101 0.0.0.0 any
exit
ip sla 1
icmp-echo 172.16.2.250 source-interface fastEthernet2/0
exit
ip sla schedule 1 life forever start-time now
ip sla 2
icmp-echo 172.16.2.251 source-interface fastEthernet2/0
exit
ip sla schedule 2 life forever start-time now
track 1 ip sla 1 reachability
track 2 ip sla 2 reachability
track 100 interface S0/0 line-protocol
route-map riverbed permit 10
match ip address Remotes-To-A
set ip next-hop verify-availability 172.16.2.250 1 track 1
set ip next-hop verify-availability 172.16.2.251 2 track 2
route-map riverbed permit 20
match ip address Remotes-To-B
set ip next-hop verify-availability 172.16.2.251 1 track 2
set ip next-hop verify-availability 172.16.2.250 2 track 1
route-map riverbed permit 30
match ip address Catch-All

```

```

set ip next-hop verify-availability 172.16.2.250 1 track 1
set ip next-hop verify-availability 172.16.2.251 2 track 2
exit
interface fa2/0
ip address 172.16.2.2 255.255.255.0
standby 1 ip 172.16.2.254
standby 1 preempt
standby 2 ip 172.16.2.253
standby 2 priority 110
standby 2 preempt
standby 2 track 100 20
exit
interface fa1/0
ip policy route-map riverbed
interface S0/0
ip policy route-map riverbed
exit
exit
write memory

```

Configuring Local PBR for ICMP Redirection in a Mixed MTU Environment

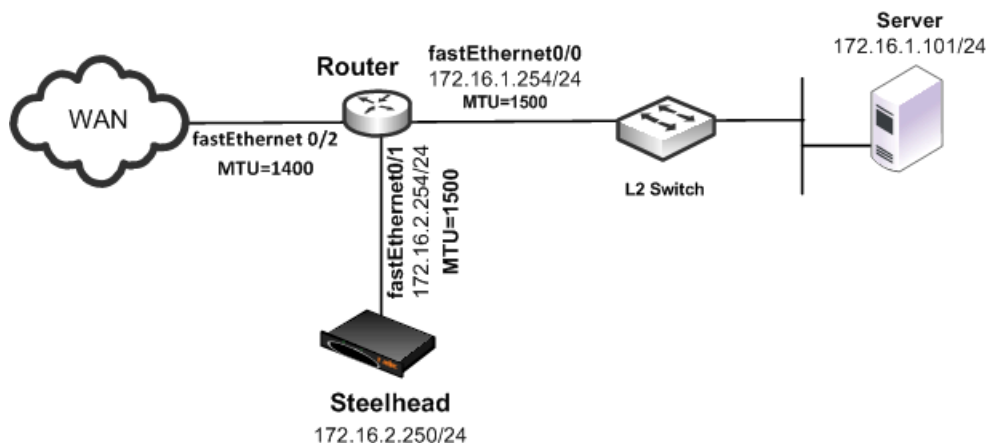
You can add a Local PBR configuration in addition to the regular PBR configuration discussed in previous sections. You use a Local PBR configuration in environments where ICMP messages generated by the router need special routing configurations: for example, mixed-size maximum transmission unit (MTU) environment with Steelhead appliance full-transparency in-path rules.

In networks that have a mix of MTU interface configurations, path MTU (PMTU) discovery determines the MTU size on the network path between two IP hosts, to avoid IP fragmentation. Network routers send *ICMP Fragment needed but do not fragment bit set* messages and packets back to the sending host; the host then decreases the segment size and retransmits the segment.

You must forward ICMP packets to the correct host (the client, server, or Steelhead). In some network scenarios, you need a Local PBR configuration to ensure ICMP messages, generated by a router, are redirected out of the same interface of the inbound IP datagram that triggers the ICMP message. Cisco Local PBR is a local policy route map configuration that can achieve this.

Figure 10-6 shows an example scenario for which you can configure Local PBR.

Figure 10-6. Local PBR Configuration



In this example, the:

- router fastEthernet0/0 interface is attached to the layer-2 switch. The MTU is configured for 1500 bytes.
- router fastEthernet0/1 interface is attached to the Steelhead appliance. The MTU is configured for 1500 bytes.
- router fastEthernet 0/2 interface is attached to the WAN. The MTU is configured for 1400 bytes.
- Steelhead appliance is configured to optimize with full transparency.

Without a Local PBR configuration on the router, optimized connections on the Steelhead can attempt to send IP datagrams of 1500 bytes across the WAN. Due to PMTU discovery, the router generates an *ICMP fragment needed but do not fragment bit set* message.

Because the Steelhead appliance is configured with full transparency, the router forwards the ICMP message back to the server instead of the Steelhead appliance. This results in the server reducing its segment size for the connection, when in fact, the Steelhead appliance must be the one to reduce its segment size. This confusion results in transfer failure.

You can use the following Local PBR configuration on the router to ensure that packets received on fastethernet0/1, and trigger any ICMP message on the router, are forwarded to the Steelhead appliance.

To configure Local PBR on the router

- On the PBR router, at the system prompt, enter the following commands:

```
Enable
Configure terminal
ip local policy route-map local_pbr_1
access-list 101 permit icmp host 172.16.2.254 any
route-map local_pbr_1 permit 10
match ip address 101
set ip next-hop 172.16.2.250
exit
exit
write memory
```

Exporting Flow Data and Virtual In-Path Deployments

In virtual in-path deployments, such as PBR, traffic moves in and out of the same WAN0_0 interface. The LAN interface is not used. When the Steelhead appliance exports data to a flow data collector, all traffic has the wan0_0 interface index, making it impossible for an administrator to use the interface index to distinguish between LAN-to-WAN and WAN-to-LAN traffic.

You can configure the fake index feature on your Steelhead appliance to insert the correct interface index before exporting data to a flow data collector.

For details, see [“Configuring Flow Data Exports in Virtual In-Path Deployments” on page 220](#).

CHAPTER 11 IPv6

Internet Protocol version 6 (IPv6) has become an important factor to consider as the pool of IPv4 addresses becomes exhausted. You must also plan to meet the organizational mandates for compliance with IPv6. This chapter describes how to configure Steelhead appliances running RiOS v8.5 for optimization of TCP-over-IPv6 using auto-discovery and fixed-target rules. The benefit of auto-discovery and fixed-target rules is that you can provide both application streamlining for select protocols and transport streamlining, in addition to data streamlining. Packet mode optimization is still an alternative for data streamlining of IPv6 traffic. This chapter includes the following sections:

- [“Overview of IPv6” on page 275](#)
- [“In-Path Rules” on page 280](#)
- [“Protocol Support” on page 287](#)
- [“Verification and Troubleshooting” on page 287](#)

Overview of IPv6

This section provides an overview of various aspects of IPv6. This section includes the following topics:

- [“RiOS RFC Compliance and Feature Compatibility” on page 276](#)
- [“IPv6 Addressing” on page 278](#)
- [“In-Path Rules” on page 280](#)
- [“Traffic Interception” on page 279](#)

IPv6 is the next revision of the Internet Protocol. IPv6 has become a critical enabler as more devices are being connected to the Internet. The main purpose of IPv6 is to help service providers and companies manage the exhaustion of IPv4 addresses without relying on other techniques such as network address translation (NAT), which hide and manipulate the source IP address when connecting across organizational boundaries. IPv6 was designed to overcome the limitation in IPv4 by using a 128-bit address instead of a 32-bit address, thereby supporting up to 3.4×10^{38} addresses; IPv4 supports up to 4.3 billion only.

The use of private IPv4 address space has created challenges for applications, security, and performance because the source and destination addresses can change through the connection. RiOS extended support for IPv6 traffic with packet mode optimization, and RiOS v8.5 or later further enhanced its IPv6 capabilities by supporting auto-discovery and fixed target rules. By using auto-discovery or fixed target rules, RiOS can apply transport and application streamlining techniques (similarly as it does for TCP connections over IPv4) to improve the user experience as the transition to IPv6 continues.

RiOS RFC Compliance and Feature Compatibility

The following RiOS v8.5 features are compatible with IPv6.

RiOS IPv6 Support Includes	RiOS Version	Notes
Conformance with Request for Comments (RFCs) 1981, 2460, 2464, 2710, 3590, 4007, 4291, 4443, 4861, 4862, 4943, 5095, and 5156	v8.5 and later	
TCP IPv6 traffic interception between source and destination, bandwidth optimization	v8.5 and later	
Auto-discovery of Steelhead appliances	v8.5 and later	TCP inner connections between the peer Steelhead appliances is strictly IPv4.
Ability to automatically discover fixed-target and pass-through in-path rules, along with ability to deny and reject IPv6 TCP traffic as configured in the in-path rules	v8.5 and later	<p>RiOS does not support the Outlook Anywhere and Citrix latency optimization policies for auto-discovery and fixed-target rules.</p> <p>RiOS does not support the neural framing modes Always, TCP Hints, and Dynamic.</p> <p>RiOS does not support the Oracle forms and Oracle forms over SSL preoptimization policies.</p>
HTTP and HTTPS latency optimization for IPv6 TCP traffic	v8.5 and later	
Ability to configure serial clusters	v8.5 and later	
Interception of IPv6 traffic for in-path, virtual in-path, and server-side out-of-path configurations	v8.5 and later	<p>WCCPv6 support is not available. Virtual in-path support is PBR only. Interceptor is not supported.</p>
Intercepting and passing through IPv4 and/or IPv6 traffic, depending on the in-path rules	v8.5 and later	
Ability to detect asymmetric routes for IPv6 TCP traffic; enables connection forwarding of IPv6 TCP traffic in asymmetric conditions	v8.5 and later	The connection-forwarding control channel between the neighbors is strictly IPv4. This requires configuring IPv4 addresses on the Steelhead appliances.
Ability to configure IPv4 and IPv6 addresses on every in-path interface and intercepting and optimizing IPv4 and IPv6 traffic	v8.5 and later	
<p>Ability to configure one IPv6 address configuration for every in-path interface</p> <p>RiOS intercepts and optimizes traffic matching the scope of the IPv6 address configured on the in-path interface. Not applicable for a link-local address configured on the in-path interface.</p>	v8.5 and later	RiOS passes through IPv6 TCP traffic not matching the scope of the IPv6 address configured on the in-path interface.
<p>Ability to configure IPv6 addresses on any in-path interface</p> <p>IPv6 TCP inner connections only in fixed-target cases</p>	v8.5 and later	This IPv6-only mode requires configuring only fixed-target in-path rules

RiOS IPv6 Support Includes	RiOS Version	Notes
Enhanced auto-discovery of Steelhead appliances for IPv6 TCP traffic	v8.5 and later	TCP inner connections between the peer Steelhead appliances is IPv4 only.
Simplified routing for IPv6 TCP traffic	v8.5 and later	
Connection forwarding for IPv6 traffic in multi-interface mode.	v8.5 and later	The control connection between neighbors is still IPv4 only. When multiple interface support on the Configure > Networking > Connection Forwarding page is not enabled, IPv6 traffic is passed through.
Ability to configure peering rules for IPv6 traffic.	v8.5	The peer client-side Steelhead appliance IP address is IPv4 only.
Ability to configure IPv6 addresses in Single Ended Interception (SEI) rules under Configure > Networking > Transport Settings	v8.5 and later	
Global and automatic kickoff for pass-through TCP IPv6 traffic	v8.5 and later	
Ability to configure asymmetric VLANs for IPv6 TCP traffic	v8.5 and later	
Latency optimization of signed-SMB, CIFS/SMB1, SMB2, and SMB3 using IPv6 endpoint addressing	v8.5.2 and later	The authentication stack continues to require IPv4 endpoint addressing
Encrypted Outlook Anywhere latency optimization	v8.6 and later	
MAPI, eMAPI latency optimization	v8.6 and later	Authentication is over IPv4.
Authentication over IPv6	v8.6 and later	

Features Not Supported with IPv6

The following features are not IPv6 compatible:

- Management In-Path (MIP) Interface
- Transparency
- NetFlow
- RSP
- Path selection
- QoS
- Host labels
- IPSec
- Automatic address assignment through DHCPv6
- Multicast listener discovery

- IPv6 stateless address auto-configuration
- WCCP using anything other than IPv4 outer connections
- Connection-forwarding neighbor connection using anything other than IPv4
- ICMPv6 redirect messages

IPv6 Addressing

IPv6 addresses share many characteristics with their IPv4 counterparts. Each version separates the address into a network identifier and host identifier. Depending on the type of address, IPv6 specifies certain bit ranges for specific purposes.

In many ways, this is similar to how IPv4 uses address classes. It is important to note that IPv6 addresses use hexadecimal digits instead of a dotted-decimal format. Also, an interface on a device has multiple IPv6 addresses (instead of a single IPv4 address) that it uses to communicate. For example, each interface always has a link-local address, and can have one or multiple global unicast addresses.

There are several types of IPv6 addresses. Riverbed recommends that you verify with the IPv6 address registry, because new IPv6 transition mechanisms or address blocks are reserved. The types of addresses are as follows:

- Global unicast--2000::/3
 - 2002::/16 reserved for 6to4
 - 2001:0000::/32 reserved for TEREDO
- ::ffff:0:0/96 reserved for IPv4-mapped address
- Unique local--fc00::/7
- Link local--fe80::/10
- Multicast--ff00::/8
- Loopback---:1/128 reserved for loopback address
- Unspecified--::/128 reserved for unspecified address

Source: <http://www.iana.org/assignments/ipv6-address-space/ipv6-address-space.xml> (April 2013)

Note: The types of IPv6 addresses are subject to change.

RiOS supports a single-system, generated link-local address and a user-assigned IPv6 address for each interface (primary, auxiliary, and in-path). The link-local address is automatically assigned as the network identifier using FE80 in the first 64 bits, and a modified EUI-64 identifier for the host bits following RFC 4291. This address is automatically assigned by the system and cannot be changed.

The link-local address is used as part of IPv6 neighbor discovery, which is analogous to address resolution protocol for IPv4. You must manually configure the user-assigned IPv6 address—it can not be derived from stateless automatic configuration or dynamic host communication protocol for IPv6. There are two address types, depending on whether you need to natively communicate outside of an organizational boundary—in much the same way a public or private IPv4 address is selected. For Steelhead appliances, choose one of the following IPv6 addresses:

- An *aggregatable global unicast* address for the Steelhead appliance to communicate with devices not directly connected to the interface.

- A *unique local unicast* address, which is the updated address range to replace site local addresses.

The intent of a unique local unicast address is for a private IPv6 address range.

The IPv6 gateway is also user assigned, and you can link it to the local address of the router or the address on the same subnet as the manually assigned address. RiOS does not support receiving router advertisements as a means to discover the IPv6 gateway. Configure the link-local address for a virtual router in circumstances in which you use a first hop redundancy protocol: for example, HSRPv6.

Traffic Interception

RiOS v8.5 introduces the ability to intercept TCP-over-IPv6 traffic and perform optimization by performing transport streamlining on the connection instead of performing data reduction on a packet-by-packet basis. The Steelhead appliances perform their roles in the connection by establishing three separate connections:

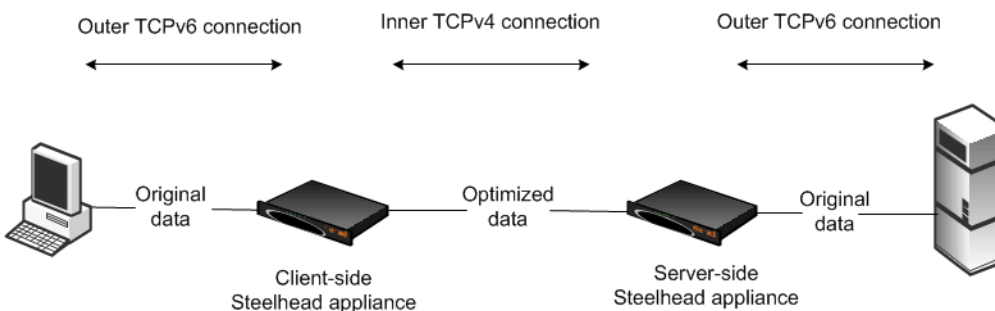
- outer channel from the client to the client-side Steelhead appliance
- outer channel from the server to the server-side Steelhead appliance
- inner channel between Steelhead appliances

The key difference between packet mode optimization and IPv6-based optimization is that IPv6 supports auto-discovery and fixed-target rules. Packet mode performs data reduction on each packet, and TCP-over-IPv6 supports application, data, and transport streamlining by inserting the Steelhead appliances into the connection.

Support for WAN visibility modes with auto-discovery of TCP-over-IPv6 traffic is limited to correct addressing. Identical to IPv4 auto-discovery, IPv6 auto-discovery uses the same TCP options present in the TCP header of the connection setup packets to discover a remote Steelhead appliance on the path between the end systems. Note that the Steelhead appliances set up an inner channel using IPv4 addresses. This requires the intervening network to be IPv6- and IPv4-capable or dual-stack.

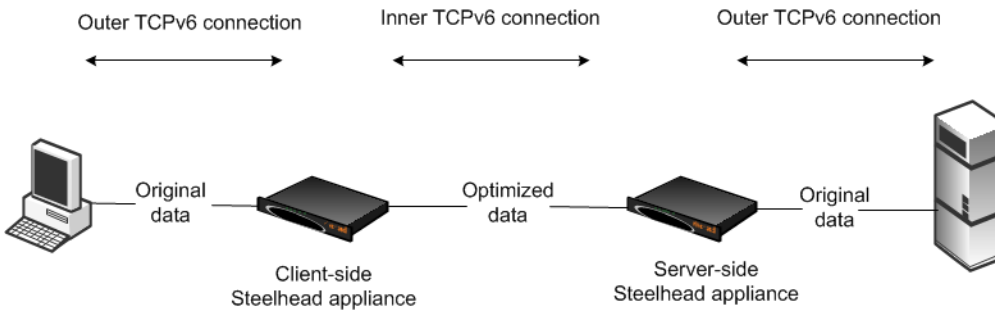
For more information about packet mode optimization, see [“Packet Mode Optimization” on page 289](#). For more information about WAN visibility modes and correct addressing, see [“WAN Visibility Modes” on page 49](#).

Figure 11-1. Auto-Discovery with IPv6



The alternative to using auto-discovery is to use a fixed-target rule. A fixed-target rule enables you to optimize traffic end-to-end using IPv6 addresses. The inner channel between Steelhead appliances forms a TCP connection using the manually assigned IPv6 address. This method is similar to an IPv4 fixed-target rule, and you configure it the same way. Note that although you can use IPv6 addresses end-to-end in the connection, the Steelhead appliance in-path interface continues to require an IPv4 address for the optimization service to start.

Figure 11-2. Fixed-Target Rule with IPv6



In-Path Rules

The default rule has changed from *all-IPv4* to *all-IP*. All-IP includes all IPv4 and all IPv6 traffic. The default rule for TCP traffic, either IPv4 or IPv6, attempts auto-discovery with correct addressing as the WAN visibility mode. A fixed-target rule with an IPv6 target appliance address requires the source and destination address type to be an IPv6 address. You must change the use of all-IP to all-IPv6. If you do not change to all-IPv6, use specific source and/or destination addresses.

Deployment Options

You can configure Steelhead appliances for in-path or virtual in-path deployment for TCP-over-IPv6 traffic. Riverbed also supports server-side out-of-path deployments. This section includes the following topics:

- [“Configuring an In-Path Steelhead Appliance IPv6 Deployment” on page 281](#)
- [“Configuring a Steelhead Appliance Serial Cluster IPv6 Deployment” on page 282](#)
- [“Configuring a Connection Forwarding and Steelhead Appliance IPv6 Deployment” on page 284](#)
- [“Configuring a Virtual In-Path Steelhead Appliance IPv6 Deployment” on page 285](#)
- [“Configuring a Fixed-Target Rule Steelhead Appliance IPv6 Deployment” on page 286](#)

Considerations for the deployment type are the same as the considerations for optimizing of IPv4 connections. Network integration features such as fail-to-wire, link state propagation, parallel deployments, firewalls, and so on continue to be relevant for optimization of TCP-over-IPv6 traffic. IPv4 connections can co-exist with TCP-over-IPv6 traffic.

The following list can help you simplify the choice of deployment options:

- Use a fixed-target-rule or packet mode optimization for all IPv6 networks. Use auto-discovery if you have a mix of IPv4 and IPv6. You must always configure an IPv4 address on the Steelhead appliance in-path interface if you are using the interface for optimization. If you do not configure the interface with an IPv4 addresses, the service does not start.
- If you have a dual-stack network, use auto-discovery.
- A parallel Steelhead appliance deployment requires IPv4 between the local Steelhead appliance in-path interfaces, for connection forwarding.
- Virtual in-path supports policy-based routing. WCCP and Interceptor appliance deployments are not supported with TCP over IPv6.

Configuring an In-Path Steelhead Appliance IPv6 Deployment

The in-path deployment for optimization of TCP-over -IPv6 traffic is similar to an in-path deployment for IPv4 connections. You deploy the Steelhead appliance physically in-path using an IPv6 address on the Steelhead appliance in-path interface. Optionally, for IPv6 management, you can configure an IPv6 address on the primary interface.

In addition to using the primary interface, you can also use the auxiliary (AUX) interface for management. The AUX interface must be on a different subnet than the primary interface. The in-path management interface can not be assigned an IPv6 address to manage the Steelhead appliance.

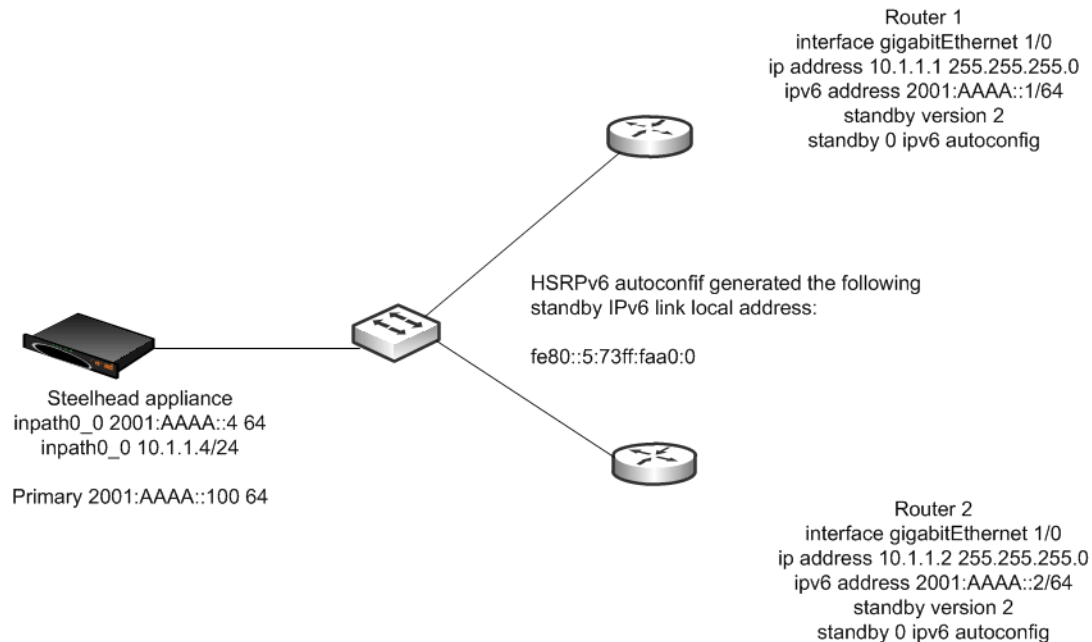
The in-path interface has its own IPv6 routing table. You can add destinations to the table and select an appropriate next hop. You can also add simplified routing to reduce the burden of administering IPv6 routes.

For details on simplified routing, see [“Overview of Simplified Routing” on page 45](#).

[Figure 11-3](#) shows an example deployment with an IPv6 address from the globally assigned address range manually assigned to the primary and inpath0_0 interface. The primary interface is assigned 2001:aaaa:100 with a prefix length of 64 bits. The inpath0_0 interface is assigned 2001:aaaa::10 with a prefix length of 64 bits. The primary interface has the Global unicast address of Router #1 configured as its IPv6 gateway, but you could use a link-local address.

In this example, the link-local address of the HSRPv6 virtual gateway is used as the default route for the in-path interface and the global unicast address is used for the primary interface. Normally, the same default route is used. The in-path interface is configured to use the HSRPv6 virtual gateway link local address for its IPv6 gateway. This link-local address is always on the same network as the Steelhead appliance in-path interface automatically assigned link-local address according to the IPv6 standards.

Figure 11-3. In-Path Steelhead Appliance Configuration Using IPv6



To configure an in-path Steelhead appliance using IPv6

- Connect to the Steelhead CLI and enter the following commands:

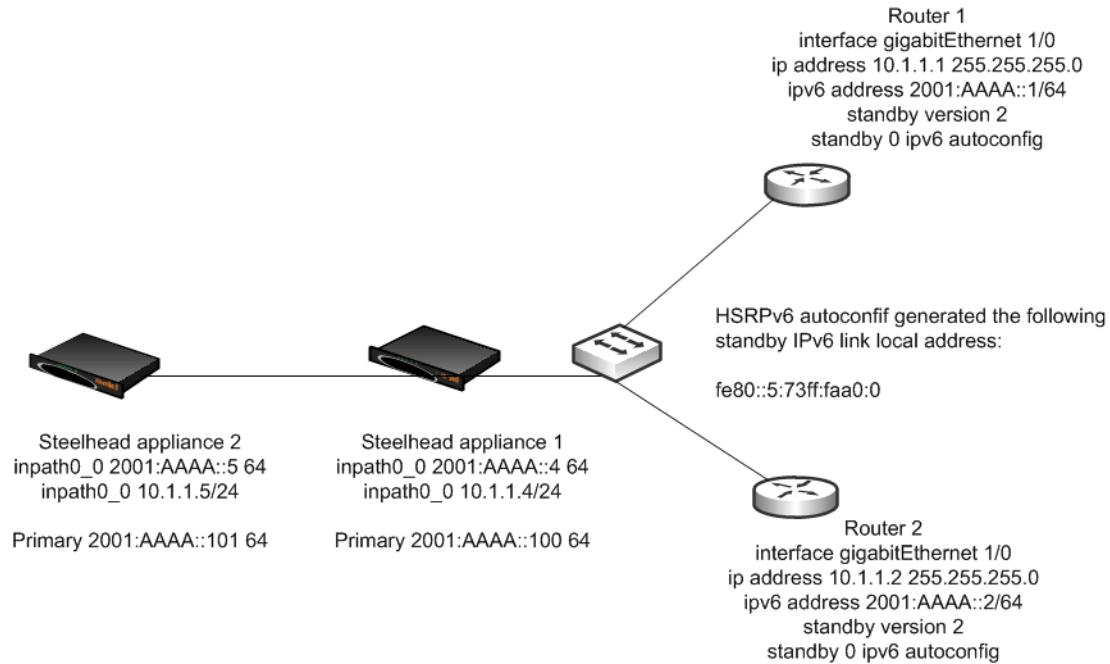
```
enable
configure terminal
interface primary ipv6 address 2001:aaaa::100 64
ipv6 default-gateway "2001:aaaa::1"
interface inpath0_0 ipv6 address 2001:aaaa::4 64
ipv6 in-path-gateway inpath0_0 fe80::5:73ff:fea0:0
in-path enable
Using Auto-Discovery requires an IPv4 address
interface inpath0_0 ip address 10.1.1.4 /24
ip in-path-gateway inpath0_0 10.1.1.1
```

Configuring a Steelhead Appliance Serial Cluster IPv6 Deployment

Figure 11-4 shows an example deployment with an IPv6 addresses in a Steelhead appliance serial cluster deployment. You can deploy Steelhead appliances in a serial cluster and optimize TCP-over-IPv6 traffic. Set the peering rule to match the peer IPv4 address, because Steelhead appliances insert the IPv4 address in the auto-discovery probe to identify themselves in the auto-discovery process. Setting the peering rule for serial cluster deployments correctly is important because Riverbed does not recommend that you optimize connections between locally connected Steelhead appliances.

For details on serial cluster deployments, see [“In-Path Redundancy and Clustering Examples”](#) on page 184.

Figure 11-4. Serial Cluster Steelhead Appliance Configuration Using IPv6



To configure a Steelhead appliance serial cluster using IPv6

1. On Steelhead A, connect to the CLI and enter the following commands:

```
enable
configure terminal
interface inpath0_0 ip address 10.1.1.4 /24
ip in-path-gateway inpath0_0 10.1.1.1
interface inpath0_0 ipv6 address 2001:aaaa::4 64
ipv6 in-path-gateway inpath0_0 fe80::5:73ff:fea0:0
in-path enable
in-path peering auto
in-path simplified routing dest-only
in-path peering rule pass peer 10.1.1.5 rulenum end
write memory
restart
```

2. On Steelhead B, connect to the CLI and enter the following commands:

```
enable
configure terminal
interface inpath0_0 ip address 10.1.1.5 /24
ip in-path-gateway inpath0_0 10.1.1.1
interface inpath0_0 ipv6 address 2001:aaaa::5 64
ipv6 in-path-gateway inpath0_0 fe80::5:73ff:fea0:0
in-path enable
in-path simplified routing dest-only
in-path peering auto
in-path peering rule pass peer 10.1.1.4 rulenum end
write memory
restart
```

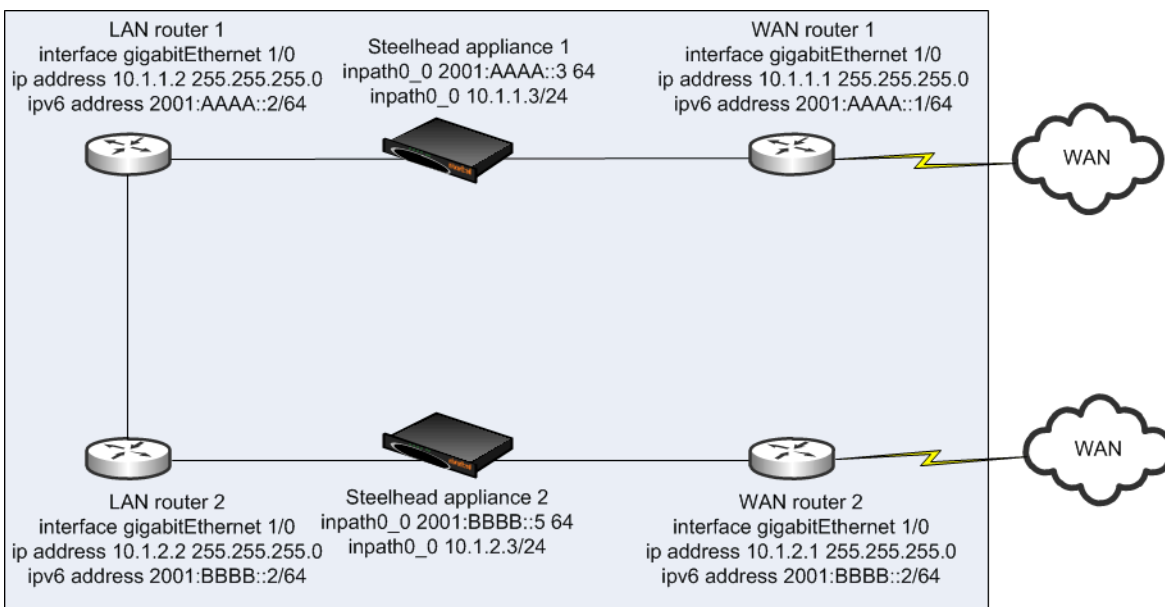
Configuring a Connection Forwarding and Steelhead Appliance IPv6 Deployment

Figure 11-5 shows an example deployment of connection forwarding for TCP-over-IPv6 traffic. You can deploy Steelhead appliances in a parallel topology using connection forwarding to optimize TCP-over-IPv6 traffic. You must use RiOS v8.5 or later and configure your Steelhead appliance to communicate using multi-interface. In addition, each in-path interface requires an IPv4 and an IPv6 address.

Connection forwarding uses the IPv4 addresses (TCP port 7850 connection) and redirects the connection setup packets through GRE encapsulation between the IPv4 addresses. When the connection is established and optimized, asymmetric traffic is redirected through NAT, destined to the peer in-path interface IPv6 address. You can optimize TCP over IPv4 and TCP-over-IPv6 traffic concurrently and the NAT entries are resynchronized between peers, even if a peer has its optimization service restarted or is disconnected.

For details on connection forwarding, see [“Connection Forwarding” on page 42](#).

Figure 11-5. Connection Forwarding and Steelhead Appliance Using IPv6



To configure connection forwarding and Steelhead appliances using IPv6

1. On Steelhead A, connect to the CLI and enter the following commands:

```
#--- Enable and configure the in-path Interfaces
in-path enable
interface inpath0_0 ip address 10.1.1.3 /24
interface inpath0_0 ipv6 address 2001:aaaa::3 64
#--- Set the default gateway for the in-path interface to be the LAN-side Layer-3 switch
ip in-path-gateway inpath0_0 10.1.1.1
ipv6 in-path-gateway inpath0_0 2001:aaaa::1
#--- Add routes to reach connection forwarding peer over the LAN router connection
ip in-path route inpath0_0 10.1.2.0 255.255.255.0 10.1.1.2
ipv6 in-path route inpath0_0 2001:bbbb::/64 2001:aaaa::2
#--- Enable enhanced auto discovery
in-path peering auto
#--- Simplified Routing destination only should be used and is on by default
#--- with new RiOS v6.x installs
in-path simplified routing dest-only
#--- Enable Connection Forwarding to neighbor 10.1.2.3
```

```

steelhead communication enable
steelhead communication multi-interface enable
steelhead name SteelheadB main-ip 10.1.2.3

```

2. On Steelhead B, connect to the CLI and enter the following commands:

```

#--- Enable and configure the in-path Interfaces
in-path enable
interface inpath0_0 ip address 10.1.2.3 /24
interface inpath0_0 ipv6 address 2001:bbbb::3 64
#--- Set the default gateway for the in-path interface to be the LAN-side Layer-3 switch
ip in-path-gateway inpath0_0 10.1.2.2
ipv6 in-path-gateway inpath0_0 2001:bbbb::1
#--- Add routes to reach connection forwarding peer over the LAN router connection
ip in-path route inpath0_0 10.1.1.0 255.255.255.0 10.1.2.2
ipv6 in-path route inpath0_0 2001:aaaa::/64 2001:bbbb::2
#--- Enable enhanced auto discovery
in-path peering auto
#--- Simplified Routing destination only should be used and is on by default
#--- with new RiOS v6.x installs
in-path simplified routing dest-only
#--- Enable Connection Forwarding to neighbor 10.1.1.3
steelhead communication enable
steelhead communication multi-interface enable
steelhead name SteelheadA main-ip 10.1.1.3

```

Configuring a Virtual In-Path Steelhead Appliance IPv6 Deployment

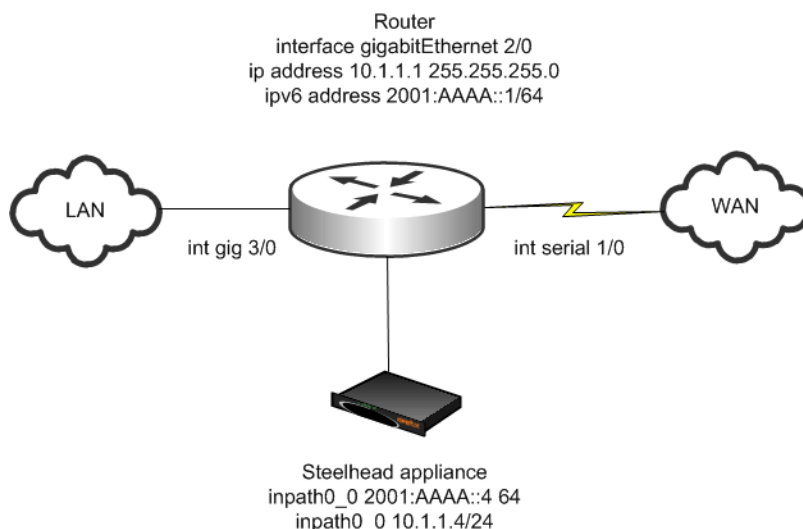
Figure 11-6 shows an example deployment of a virtual in-path Steelhead appliance using IPv6. Virtual in-path support for TCP-over-IPv6 traffic is limited to policy-based routing (PBR) in RiOS v8.5 or later. You configure a Steelhead appliance virtually in-path similarly to an IPv4 PBR deployment.

WCCPv6 is not supported. The Layer-3 device redirecting traffic must support PBR for IPv6 traffic. Cisco provides a list of software in which the PBR for IPv6 feature was first introduced (12.2(30)S, 12.2(33)SX14, 12.3(7)T, 12.4, and 12.4(2)T).

Source: http://www.cisco.com/en/US/docs/ios/ipv6/configuration/guide/ip6-pol-bsd_rtng.html

For details on PBR, see “Policy-Based Routing Virtual In-Path Deployments” on page 255.

Figure 11-6. Virtual In-Path Steelhead Appliance Configuration Using IPv6



To configure a virtual in-path Steelhead appliance using IPv6

1. Connect to the Steelhead CLI and enter the following commands:

```
interface inpath0_0 ipv6 address 2001:aaaa::4 64
ipv6 in-path-gateway inpath0_0 2001:aaaa::1
in-path enable
in-path simplified routing "none"
in-path oop enable
```

You must configure an IPv4 address for auto-discovery of TCP-over-IPv6 traffic if you are using connection forwarding.

2. Configure the Layer-3 device (use Cisco IOS syntax):

```
ipv6 access-list OPTIMIZEv6
  permit tcp any host 2001:DDDD::100
  permit tcp host 2001:DDDD::100 any
route-map OPTIMIZEv6 permit 10
  match ipv6 address OPTIMIZEv6
  set ipv6 next-hop 2001:aaaa::4
interface serial1/0
  ipv6 policy route-map OPTIMIZEv6
interface gigabitEthernet3/0
  ipv6 policy route-map OPTIMIZEv6
```

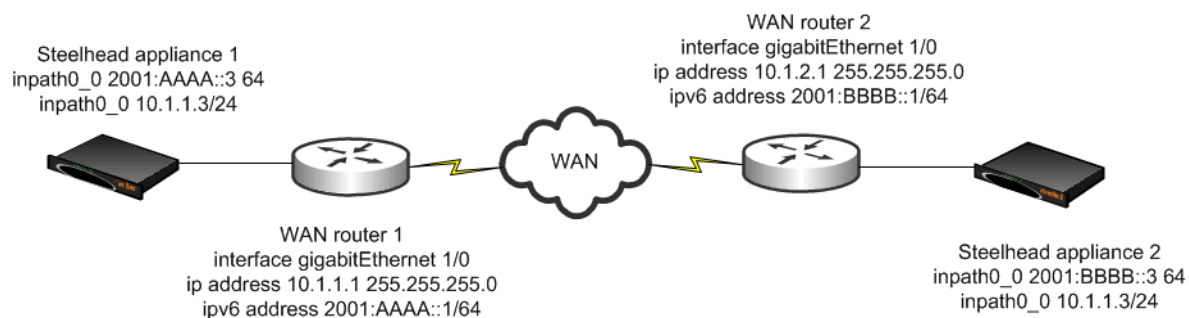
Configuring a Fixed-Target Rule Steelhead Appliance IPv6 Deployment

Figure 11-7 shows an example deployment of a Steelhead appliance using a fixed-target rule in an IPv6 environment. You can use fixed-target rules for end-to-end optimization using an IPv6 address when you have Steelhead appliances in an all-IPv6 network, or when peer Steelhead appliances communicate using their in-path interface IPv6 addresses.

The fixed-target rule operates similarly to IPv4 traffic, by replacing the IPv4 header and sending a directed connection setup packet to the target appliance IPv6 address. A fixed-target rule also works for server-side out-of-path (SSOOP) deployments. In a SSOOP deployment, the outer channel between the server-side Steelhead appliance and server uses the primary interface IPv6 address in the same way it operates for SSOOP on IPv4 traffic.

For details on SSOOP, see [“Out-of-Path Deployments” on page 335](#).

Figure 11-7. Fixed-Target Rule Steelhead Appliance Configuration Using IPv6



To configure a Steelhead appliance using a fixed-target rule and IPv6

- Connect to the Steelhead CLI and enter the following commands:

```
enable
configure terminal
```



```

interface inpath0_0 ip address 10.1.1.3 /24
ip in-path-gateway inpath0_0 10.1.1.1
interface inpath0_0 ipv6 address 2001:aaaa::3 64
ipv6 in-path-gateway inpath0_0 2001:aaaa::1
in-path enable
in-path simplified routing dest-only
in-path peering auto

in-path rule fixed-target target-addr 2001:bbbb::3 target-port 7810 backup-addr :: backup-port
7810 dstaddr all-ipv6 dstport "80" srcaddr all-ipv6 preoptimization "none" optimization
"normal" latency-opt "normal" neural-mode "always" wan-visibility "correct" vlan -1 description
"" auto-kickoff disable rule-enable true rulenum start

```

Protocol Support

Optimization for TCP-over-IPv6 traffic supports data reduction for any connection. In addition to data reduction, the following protocols have application streamlining support:

- FTP
- SSL
- HTTP

The configuration for FTP and SSL is transparent and you do not need to perform any configuration.

SSL optimization shows a display for discovered SSL servers (you can optimize) and another display for discovered servers (bypassed; you cannot optimize for SSL). The displays show the IPv6 address of the server.

HTTP enables you to specify settings for a particular server or subnet using an IPv6 address similar to how you configure HTTP for IPv4 addresses. The following shows an example from the CLI:

```

protocol http server-table subnet 2001:dddd::100/128 obj-pref-table no parse-prefetch no url-
learning no reuse-auth no strip-auth-hdr no gratuitous-401 no force-nego-ntlm no strip-compress no
insert-cookie yes insrt-keep-aliv no FPSE no WebDAV no FSSHTTP no

```

For more information about FTP, SSL, and HTTP, see the *Steelhead Appliance Deployment Guide - Protocols* and the *Riverbed Command-Line Interface Reference Manual*.

Verification and Troubleshooting

There are three ways to verify and troubleshoot your IPv6 deployments:

- Utility ping
- Traceroute
- Connection report

Make sure that the Steelhead appliance can reach the end system (client or server). You can use a utility ping at the Steelhead CLI and in the Steelhead Management Console. Using the ping6 utility, you must specify the interface IPv6 address instead of the interface name. The following example sends an ICMPv6 echo request using the IPv6 address of 2001:aaaa::10 to reach 2001:aaaa::2:

```

VSH # ping6 -I 2001:aaaa::10 2001:aaaa::2
PING 2001:aaaa::2(2001:aaaa::2) from 2001:aaaa::10 : 56 data bytes
64 bytes from 2001:aaaa::2: icmp_seq=1 ttl=64 time=22.0 ms

```

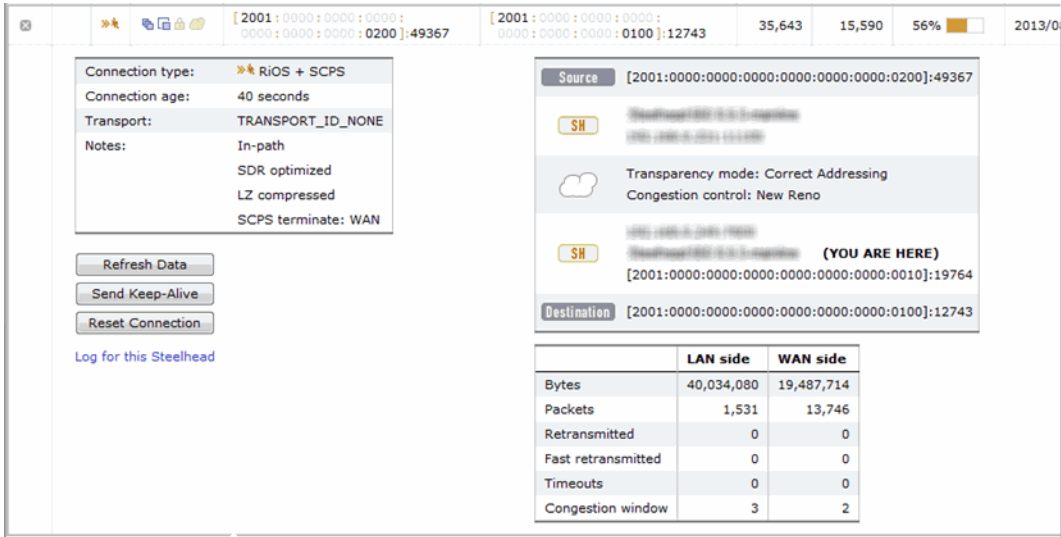
```
64 bytes from 2001:aaaa::2: icmp_seq=2 ttl=64 time=31.1 ms
64 bytes from 2001:aaaa::2: icmp_seq=3 ttl=64 time=20.3 ms
```

Traceroute for IPv6 has also been included for verifying the path from the Steelhead appliance to an IPv6 device or verifying the path between Steelhead appliances using a fixed-target rule:

```
VSH # traceroute6 -I 2001:aaaa::10 2001:bbbb::2
traceroute to 2001:aaaa::10 (2001:aaaa::10), 30 hops max, 2001 byte packets
 1  2001:aaaa::10 (2001:aaaa::10)  0.328 ms  309.442 ms  309.450 ms
```

Figure 11-8 shows a connection report in which the status of an IPv6 connection is optimized by the Steelhead appliance.

Figure 11-8. Connection Report



CHAPTER 12 Packet Mode Optimization

This chapter describes packet mode optimization. Packet mode optimization enables the Steelhead appliance to perform packet-by-packet traffic optimization on various types of IPv4 and IPv6 traffic. This chapter includes the following sections:

- [“Comparison with TCP Proxy Mode Optimization” on page 289](#)
- [“Configuring Packet Mode Optimization” on page 290](#)
- [“Design Considerations” on page 294](#)
- [“Best Practices for Packet Mode Optimization” on page 295](#)

This chapter requires that you be familiar with [“Optimization Techniques and Design Fundamentals” on page 7](#), including data streamlining and fixed-target in-path rules.

For more information about IPv6, see [“IPv6” on page 275](#).

Overview of Packet Mode Optimization

In RiOS v7.0 or later, the packet mode optimization feature can optimize TCP IPv6 and UDP IPv4 traffic. When you use packet mode optimization, the Steelhead appliance applies the same SDR and LZ data streamlining techniques to UDP or TCP IPv6 packets. Packet mode optimization achieves data reduction rates similar to TCP IPv4 traffic.

RiOS v8.5 or later expands support of packet mode optimization to include TCP IPv4 and UDP IPv6 traffic. In addition, connection or flow reporting for packet mode optimization is greatly enhanced. To optimize TCP IPv4 or UDP IPv6, the client-side and server-side Steelhead appliances must run RiOS v8.5.

Comparison with TCP Proxy Mode Optimization

The bulk of most network traffic is TCP IPv4 traffic, which Steelhead appliances typically optimize using a TCP proxy architecture. TCP proxy mode optimization separates a TCP connection into three individual connections:

- Client to client-side Steelhead appliance
- Steelhead appliance to Steelhead appliance

- Server-side Steelhead appliance to server

The advantage of TCP proxy mode is that it increases performance, because the Steelhead appliance acts a local proxy to the host. This enables the Steelhead appliance to perform transport streamlining and application streamlining, which results in increased user performance.

However, sometimes you might want to use the Steelhead appliance optimization to reduce the amount of traffic traversing the WAN. Packet mode optimization provides a simple approach in which the Steelhead appliance looks at a packet, or small group of packets, and performs SDR and LZ on the data payload for data reduction. The host and Steelhead appliance do not create an individual TCP handshake, and the Steelhead appliance reduces payload for packets as the traffic flows through.

The advantage of packet mode optimization is that it is a universal method that applies data streamlining to diverse protocols. The disadvantage is the lack of performance benefits from transport streamlining or application streamlining, because the Steelhead appliance does not proxy or perform intelligent application prediction.

Packet mode optimization is unidirectional—from sender to receiver—which is an effect of not acting as a proxy. For example, take two sites with Steelhead appliances, Site A and Site B. A device at Site A sends SNMP traps over UDP to a server at Site B, and only Steelhead appliance A is configured to optimize UDP traffic to Steelhead appliance B. If the server at Site B, for some reason, responds over UDP to the device at Site A, this traffic is not optimized unless you have configured Steelhead B to optimize UDP traffic to Steelhead A.

Configuring Packet Mode Optimization

This section describes how to configure packet mode optimization for UDP IPv4 and TCP IPv6 traffic.

Packet mode optimization supports only correct addressing. Packet mode optimization does not support auto-discovery and requires that you configure fixed-target, packet-mode, in-path rules.

For more design details, see [“Design Considerations” on page 294](#).

Note: the following example shows UDP traffic.

To configure packet mode optimization for IPv4 traffic

1. From the Management Console, choose Configure > Optimization > General Service Settings and select Enable Packet Mode Optimization; otherwise, use the CLI command **packet-mode enable**.

Figure 12-1. Enable Packet Mode Optimization

Configure > Optimization > General Service Settings ?

In-Path Settings

- ☒ Enable In-Path Support
 - ☐ Reset Existing Client Connections on Start Up *(not recommended for production networks)*
 - ☐ Enable L4/PBR/WCCP/Interceptor Support
 - ☒ Enable Optimizations on Interface **inpath0_0**

Out-of-Path Settings

- ☐ Enable Out-of-Path Support *(server-side appliances only)*

Connection Settings

Half-Open Connection Limit per Source IP:

Maximum Connection Pool Size:

Failover Settings

- ☐ Enable Failover Support
 - Current Appliance is:
 - IP Address (peer In-Path interface):

Packet Mode Optimization Settings

- ☐ Enable Packet Mode Optimization

Apply

2. Configure a fixed-target (packet mode optimization) in-path rule on the initiating Steelhead appliance. To optimize traffic in both directions, you must configure a similar in-path rule on the peer Steelhead appliance.

Figure 12-2 shows an example that creates an in-path rule with the following key settings.

Field	Option	Description
Type	Fixed-target (packet mode optimization)	Rule type for performing packet mode optimization.
Source subnet	all-IPv4	Selects all IPv4 addresses. You can choose specific IP addresses or ranges.
Destination subnet	all-IPv4	Selects all IPv4 addresses. You can choose specific IP addresses or ranges.
Protocol	UDP	Selects optimization of TCP, UDP, or any type of traffic.
Target appliance IP address	10.32.9.211	Specifies the remote Steelhead appliance to optimize to. You must specify a target with a fixed-target rule. You can also specify a backup appliance. A backup Steelhead appliance is not specified in this example.
Data reduction policy	Normal	Specifies SDR, LZ, or both for data reduction. Normal usage includes both.
Position	1	Determines which position the rule is in the rule list. You can decide this based on your environment.

Figure 12-2. Fixed-Target (Packet Mode Optimization) Rule for UDP IPv4 Traffic

Configure > Optimization > In-Path Rules ?

▼ Add a New In-Path Rule — Remove Selected Rules ⇅ Move Selected Rules...

Type: Fixed-Target (Packet Mode Optimization)

Source Subnet: all-IPv4 Port or Port Label: all

Destination Subnet: all-IPv4 Port or Port Label: all

VLAN Tag ID: all

Protocol: UDP

Target Appliance IP Address: 10.32.9.211 Port: 7800

Backup Appliance IP Address: Port: 7810

Data Reduction Policy: Normal

Position: End

Description: Packet Mode - UDP IPv4 Traffic

Enable Rule: ☒

Add

3. Click **Add**.

To optimize IPv6 traffic, you must enable packet mode optimization and create a fixed-target packet-mode rule similar to IPv4 traffic, but you must make sure to enable IPv6 (IPv6 is enabled by default) and configure an IPv6 setting on the Steelhead appliance in-path interfaces.

Note: This example shows TCP traffic.

To configure packet mode optimization for IPv6 traffic

1. From the Management Console, choose Configure > Optimization > General Service Settings; otherwise use the CLI command **packet-mode enable**.
2. Configure a fixed-target (packet mode optimization) in-path rule on the initiating Steelhead appliance.
To optimize TCP traffic in both directions, you must configure a similar in-path rule on the peer Steelhead appliance.

Figure 12-3 shows an example that creates a rule with the following key settings.

Field	Option	Description
Type	Fixed-target (packet mode optimization)	Rule type for performing packet mode optimization.
Source subnet	all-IPv6	Selects all IPv6 addresses. You can choose specific IP addresses or ranges.
Destination subnet	all-IPv6	Selects all IPv6 addresses. You can choose specific IP addresses or ranges.
Protocol	TCP	Selects optimization of TCP, UDP, or any type of traffic.
Target appliance IP address	10.32.9.211	Specifies the remote Steelhead appliance to optimize to. You must specify a target with a fixed-target rule. You can also specify a backup appliance. A backup Steelhead appliance is not specified in this example.
Data reduction policy	Normal	Specifies SDR, LZ, or both for data reduction. Normal usage includes both.
Position	1	Determines which position the rule is in the rule list. You can decide this based on your environment

Figure 12-3. Fixed-Target (Packet Mode Optimization) Rule for TCP IPv6 Traffic

The screenshot shows the 'Configure > Optimization > In-Path Rules' window. At the top, there are buttons: 'Add a New In-Path Rule', 'Remove Selected Rules', and 'Move Selected Rules...'. The main configuration area includes the following fields:

- Type:** Fixed-Target (Packet Mode Optimization) (dropdown)
- Source Subnet:** all-IPv6
- Destination Subnet:** all-IPv6
- VLAN Tag ID:** all
- Protocol:** TCP (dropdown)
- Target Appliance IP Address:** 10.32.9.211
- Backup Appliance IP Address:** (empty)
- Port or Port Label:** all (for both Source and Destination)
- Port:** 7800 (for Target) and 7810 (for Backup)
- Data Reduction Policy:** Normal (dropdown)
- Position:** End (dropdown)
- Description:** Packet Mode - TCP IPv6 Traffic
- Enable Rule:** ☒

An 'Add' button is located at the bottom left of the configuration area.

3. Click **Add**.
4. Connect to the Riverbed CLI. You can also use the Steelhead Management Console.
5. Enable IPv6.
IPv6 is enabled by default, unless you are using versions of RiOS prior to v8.0.
6. Configure an IPv6 address on a Steelhead appliance in-path interface: **interface inpathX_Y ipv6 address <addr> <masklen>**.
7. Add IPv6 routes: **ipv6 in-path route inpathX_Y <prefix> <prefix len> <nexthop v6 addr>**.
For example, `ipv6 in-path route inpath0_0 0::0/0 ba5e:ba11:bab3:f005:ba11:bab3:dead:bea7` sets a default IPv6 gateway.

Design Considerations

Packet mode optimization does not support several network integration features available with TCP proxy optimization. The following are limitations to packet-mode optimized traffic.

Packet mode optimization in RiOS v8.5 or later supports:

- only fixed-target (packet mode optimization). Auto-discovery is not supported.
- only correct addressing. Full transparency and port transparency are not supported. Because full transparency is not supported, VLAN transparency is also not supported.

- physically and virtually in-path deployments. The exception is WCCP for IPv6 traffic, because the Steelhead appliance WCCPv2 implementation currently does not support IPv6.
- data reduction for all IPv4 and IPv6 traffic.
- connection reporting of packet-mode optimized traffic flows.

Packet mode optimization does not support:

- RSP data flow rules.
- NetFlow export of the packet-mode optimized traffic.
- connection forwarding. This does not imply that parallel Steelhead appliance deployments do not work. As each flow is optimized unidirectionally, this means that asymmetry does not have the same relevance on packet-mode optimized traffic.
- Interceptor appliance deployments.
- server-side out-of-path configuration.
- QoS shaping, enforcement, or marking. All packet-mode optimized traffic matches the default QoS default rule and class. The exception is UDP IPv4 traffic, which you can place into an MX-TCP class.

For more information about MX-TCP class, see [“MX-TCP” on page 93](#).

Best Practices for Packet Mode Optimization

Riverbed recommends that you use the following best practice guidelines when using packet mode optimization:

- **Target specific applications and servers for packet mode optimization** - Similar to the considerations when performing optimization on TCP IPv4 traffic, certain traffic types do not lend themselves well to data reduction. For example, encrypted or compressed traffic does not receive significant data reduction. Riverbed recommends that you use rules targeting specific networks or ports instead of using broad All-IP targets. Traffic types such as TFTP or UDP-based replication traffic are example target applications.
- **Do not optimize voice or video bearer traffic** - While VoIP is one of the more pervasive applications over UDP, VoIP and video RTP traffic are already compressed using specialized codecs. If you attempt to perform further data reduction with SDR or LZ, it is ineffective and can add latency resulting in jitter.
- **Use TCP proxy mode optimization** - For application latency improvements for TCP IPv4 or IPv6 traffic, you can use the typical TCP proxy mode optimization. RiOS v8.5 or later includes TCP proxy mode optimization for IPv6 traffic.

For details on IPv6 traffic, see [“IPv6” on page 275](#).

CHAPTER 13 Satellite Optimization

This chapter describes how to configure transport optimization for satellite networks. When you tune transport settings for satellite networks, you can achieve improved performance and interoperability with other TCP performance-enhancing proxies (TCP-PEP). Tuning transport settings for satellite networks falls into two primary categories:

- **Space communications protocol specification (SCPS) discovery** - provides interoperations with third-party TCP-PEP devices.
- **Transport settings** - provide flexibility for TCP optimization algorithms.

This chapter includes the following sections:

- [“Overview of Satellite Networks” on page 297](#)
- [“Overview of SCPS” on page 299](#)
- [“TCP Optimization for Satellite Environments” on page 301](#)
- [“Licensing SCPS on a Steelhead Appliance” on page 306](#)
- [“Configuring Satellite Optimization Features” on page 307](#)
- [“Verification and Troubleshooting” on page 315](#)

Important: If you are using a release prior to RiOS v7.0, SCPS was available through an RSP package and was not native to RiOS. The RSP SCPS package interoperates with SCPS native to RiOS v7.0. RSP SCPS package licenses are not valid for use as native RiOS SCPS licenses. Contact Riverbed Support or your sales team for assistance in converting RSP SCPS package licenses to native RiOS SCPS licenses.

Note: In RiOS v7.0.1 or later, RSP is replaced with VSP. VSP comes preinstalled in the Steelhead EX. For more information about VSP, see the *Steelhead Appliance Management Console User's Guide* for the Steelhead EX. Your existing RSP packages work on VSP.

Overview of Satellite Networks

This section introduces satellite networks. This section includes the following topics:

- [“Impact of Latency” on page 298](#)

- [“Impact of Loss” on page 298](#)
- [“Satellite Transport Options” on page 299](#)

You can use satellite networks for WAN connectivity for people in remote locations or when users are in a temporary or mobile environment (for example, a ship or an oil rig). Satellite networks have several characteristics that differ from terrestrial networks. The most prevalent differences include the following:

- High latency
- Dynamic bandwidth
- Asymmetric bandwidth capability
- Loss due to signal to noise issues

These characteristics can create challenges for traditional TCP algorithms for various reasons. This chapter helps you to understand and address these characteristics.

Impact of Latency

When it comes to TCP, latency is the enemy. The higher the latency, the longer it takes ordinary TCP to ramp up and use the available capacity. In satellite networks, latency is typically 10 to 100 times higher than in terrestrial links. Latency in a single-hop satellite network usually varies from 550ms to 800ms. Dual-hop satellite network latency commonly ranges from 1100ms to 1600ms. With INMARSAT BGAN networks (broadband satellite communications) it is not unusual to observe peak latencies of 2000ms and higher when congestion is present.

Even though the Steelhead appliance has TCP optimization enabled by default, tuning transport settings for a target satellite environment results in even better performance.

Impact of Loss

Satellite communications are known for having less than perfect packet delivery. Loss can wreak havoc on TCP throughput. If the loss is high enough, it can cause sessions to time-out. Loss can happen for several reasons on satellite networks. You must determine the primary cause of loss in your network so that you can tune your TCP optimization accordingly.

The two primary reasons for loss in satellite networks are congestion and poor signal quality. Do not ignore the possibility of a simple bad cable or speed and duplex mismatch when troubleshooting. Congestion-based loss is normal and expected when a link or path is saturated. Congestion-based loss triggers a TCP stacks congestion avoidance algorithm so that individual TCP sessions can adapt intelligently. Poor signal quality increases the bit error rate (BER), which is not due to congestion, but rather because of satellite communications. BER is independent of load on the circuit.

Many different events can trigger increased BER. A few of the most common are obstructions (buildings, bridges, and so on), a misaligned satellite dish, and weather. Weather events are temporary and go away in time. With obstructions or poor angles, the loss can be more persistent. Most satellite modems have forward error correction (FEC) built-in for free, and can overcome an increased BER in many situations.

In some cases, the BER loss is too severe for the satellite modem to overcome with FEC. In these situations, you might detect consistent TCP loss. Ideally, you should solve the root cause of the increased BER, but as a network administrator, this might be out of your control. In this situation, you can leverage a TCP Stack optimized for a high error environment.

Satellite Transport Options

You can choose from many satellite transport options. Their performance characteristics vary widely. A reasonable estimation is the more terminals sharing a segment, the higher the possibility for variable latency and loss. This section provides a brief summary of popular satellite solutions, along with general assumptions of loss and latency expectations. You should verify these details for your network because they might be different than those in the following table.

Satellite Transport Option	Variable or Fixed Throughput	Latency Characteristics	Loss Characteristics
INMARSAT	Fixed	Predictable	
INMARSAT BGAN	Highly variable—shared broadband	Highly variable	Highly variable
Single channel per carrier (SCPC)	Fixed	Predictable	
TDMA	Variable	Variable due to wait time for frequency slot	
FDMA	Variable	Variable due to wait time for frequency slot	
DVB-RCS/MF-TDMA	Variable	Variable	Variable: certain codec modulations are adaptive and adjust to deal intelligently with poor signal quality.

Overview of SCPS

This section describes SCPS. It does not cover the implementation of SCPS in the Steelhead appliance. This section includes the following:

- [“SCPS Benefits” on page 300](#)
- [“Common Uses for SCPS” on page 300](#)
- [“SCPS and Steelhead Appliances” on page 300](#)

For details on how to implement SCPS and the Steelhead appliance, see [“TCP Optimization for Satellite Environments” on page 301](#).

SCPS is a group of several protocol specifications developed by the consultative committee for space data systems (CCSDS) to address the limitations of communications in space. In the WAN optimization market, SCPS refers to the transport protocol specification, otherwise known as SCPS-TP. SCPS-TP is the most widely supported SCPS protocol. In the WAN optimization market, SCPS-TP is commonly called SCPS. The use of SCPS in Steelhead appliances is specifically referencing SCPS-TP. Definitions for all SCPS protocols are as follows:

- **SCPS-FP (file transfer protocol)** - A set of extensions to FTP to make it more bit efficient and to add advanced features such as record update within a file and integrity checking on file transfers. This is an optional protocol.

- **SCPS-TP (transport protocol)** - A set of transmission control protocol (TCP) options, such as selective negative acknowledgment (SNACK), selective acknowledgment (SACK), modified slow start algorithms, modified congestion avoidance algorithms, and Windows scaling. Additionally, **SCPS-TP** includes sender-side modifications to enhance the TCP performance in the stressed environments, such as long delays, high bit error rates, and significant asymmetries. For flow negotiation, SCPS uses TCP options that are registered with the Internet Assigned Numbers Authority (IANA). As a result, the SCPS-TP stack is compatible with other recognized TCP implementations.
- **SCPS-SP (security protocol)** - Based on security protocol 3 (SP3) and network layer security protocol (NLSP), with reduced protocol overhead of header. SCPS-SP also provides integrity, confidentiality, authentication, and access control for the data transmitted over the network. SCPS-SP is an optional protocol, and is comparable to Internet Protocol Security (IPSec).
- **SCPS-NP (network protocol)** - A bit-efficient network protocol that is analogous to IP. However, it is not compatible with IP. The protocol is designed for use in space systems. The protocol supports static or dynamic routing, precedence, and multiple routing options. This is an optional protocol.

For more details on SCPS-TP, go to
<http://public.ccsds.org/publications/archive/714x0b2.pdf>.

SCPS Benefits

SCPS provides improved TCP performance in high-latency and high-loss environments, such as satellite networks. As a specification, SCPS enables third-party WAN optimization solutions that support SCPS, to provide TCP acceleration in a heterogeneous WAN optimization environment.

SCPS is designed for use in dual-ended proxy (symmetric) scenarios. However, some SCPS devices also provide single-ended optimization. This enables a device to provide sender-side benefits even when there is no optimization appliance at the far end. However, not all SCPS solutions support single-ended proxy implementations.

Common Uses for SCPS

The three primary markets for SCPS are as follows:

- Space agency space networks
- Commercial satellite networks
- Private government satellite networks

SCPS is a common request in large multiagency government satellite architectures in which a lowest common denominator approach to TCP acceleration is desirable. This enables the hub organization to provide basic TCP optimization to multiple agencies connecting into their teleports, without requiring a specific vendor solution.

SCPS and Steelhead Appliances

SCPS is available in RiOS v7.0 or later. An SCPS license enables Steelhead appliances running RiOS v7.0 to negotiate SCPS optimization with another Steelhead appliance or a third-party WAN optimizer or TCP-PEP. The SCPS license also enables the SCPS per-connection and SCPS error-tolerance transport setting options. These two TCP stacks are tuned specifically for satellite networks. Steelhead appliances running RiOS v8.5 or later include a rate pacing mechanism to further tune the TCP. The mechanism can also negotiate compression with a third-party WAN optimizer or TCP-PEP.

You can configure transport optimization used by the Steelhead appliance separately from SCPS negotiation. This separation provides extensive flexibility for a broad range of environments.

A Steelhead appliance with SCPS, optimizing traffic to another Steelhead appliance, uses the configured TCP stack and still performs standard RiOS optimization functions such as data reduction and application streamlining. The same Steelhead appliance can negotiate SCPS with third-party TCP-PEP devices. This provides organizations with an approach for supporting third-party SCPS interoperability, while at the same time maximizing performance and productivity within their architectures with RiOS optimization.

For more details on using SCPS in Steelhead appliances, see [“TCP Optimization for Satellite Environments” on page 301](#) and [“Licensing SCPS on a Steelhead Appliance” on page 306](#).

Note: In RiOS v7.0 or later, you must license SCPS and restart the optimization service to enable SCPS negotiation service. This is irrelevant of which transport optimization method you select (for example, standard TCP, high-speed TCP, BW estimation, per connection, and error tolerance).

TCP Optimization for Satellite Environments

This section introduces TCP optimization for satellite environments. This section includes the following topics:

- [“SCPS Discovery” on page 302](#)
- [“Transport Optimization for Satellite Environments” on page 302](#)
- [“Configuring Automatic Detect TCP Optimization” on page 305](#)
- [“Integrating the Steelhead Appliance with Existing Satellite Modem TCP Acceleration” on page 306](#)

TCP optimization enhancements in RiOS v7.0 or later provide a robust and easy-to-use set of TCP optimization options for satellite networks. These mechanisms are specifically tuned for the performance challenges of satellite environments. The capabilities in RiOS v7.0 or later fall into two primary categories:

- SCPS discovery mechanisms
- Transport optimization mechanisms

To use SCPS discovery or the SCPS transport options, you must install an SCPS license on the Steelhead appliance. All non-SCPS options described in this chapter are included with the basic RiOS licenses, which include bandwidth estimation transport optimization and error recovery mode.

In RiOS v7.0 or later, the SCPS discovery advertises support for SCPS negotiation. SCPS discovery does not determine the TCP stack used during optimization. The transport optimization setting you configure on the Steelhead appliance determines the TCP stack used during optimization. This is a different from the RiOS v6.5 implementation of SCPS in RSP, which always uses SCPS transport optimization. The new approach in RiOS v7.0 provides more flexibility for complex dynamic environments, and is also easier to use.

The separation of SCPS discovery and the transport optimization setting is a subtle point, but it is important to understand when you implement Steelhead appliances within a satellite environment with SCPS requirements. The remaining subsections provide more details on SCPS discovery and transport optimization settings.

SCPS Discovery

This section describes SCPS discovery, also known as SCPS negotiation. SCPS discovery uses TCP options for discovery. SCPS uses a 4-byte TCP option with the decimal number 20 (hexadecimal 0x14). SCPS header also has extended capabilities that use a 10-byte TCP option with the decimal number 20 (hexadecimal 0x14). When you use the extended SCPS header, it immediately follows the mandatory 4-byte SCPS header. Steelhead appliances licensed for SCPS support dual-ended proxy connections to Steelhead appliances, and dual-ended proxy connections to third-party SCPS TCP-PEP devices. Keep in mind that the congestion-avoidance algorithm determines how TCP is optimized; SCPS discovery is only used for negotiating SCPS interoperability.

Client-side Steelhead appliances initially mark the SYN packet with the RiOS discovery TCP option (decimal 76 or 78). If no RiOS discovery response is detected in the initial SYN/ACK, then an RST packet is sent and a new SYN is sent with an SCPS TCP option. The new SYN uses the same client and server ports as the original SYN but has a different TCP sequence number. If an SCPS TCP option is returned in the SYN/ACK from the server-side peer, SCPS optimization is established.

If no SCPS TCP option is returned from the server side, a client-side Steelhead appliance using SCPS does not optimize the flow and passes the traffic through. If you do not want SCPS optimization for specific traffic, you can use the single-ended connection rule table to exclude it. The following table summarizes the discovery process used in the various device scenarios.

Device Location/Type	Server - Steelhead appliance without SCPS	Server - Steelhead appliance with SCPS	Server - SCPS TCP-PEP	Server - No Device
Client - Steelhead appliance without SCPS	RiOS	RiOS	No acceleration	No acceleration
Client - Steelhead appliance with SCPS	RiOS	RiOS	SCPS	No acceleration
Client - SCPS TCP-PEP	No acceleration	SCPS	SCPS	No acceleration

Transport Optimization for Satellite Environments

RiOS v8.5 or later has a broad set of transport options that you can use to adapt the TCP optimization for specific segments of your organization and respective performance characteristics. This section specifically addresses transport optimization in satellite networks. This section includes the following topics:

- [“Bandwidth Estimation” on page 303](#)
- [“Configuring Error Recovery” on page 303](#)
- [“SCPS Per Connection” on page 304](#)
- [“SCPS Error Tolerance” on page 304](#)
- [“SCPS Rate Pacing” on page 304](#)
- [“SCPS Single-Ended Rules” on page 304](#)
- [“SCPS Compression” on page 305](#)

RiOS v8.5 or later supports four specific transport settings for satellite networks. These include three TCP stacks and an error recovery mechanism. The transport settings are included in the following:

- Bandwidth estimation and error recovery are available in the base license of RiOS v6.5 or later.
- SCPS per connection and SCPS error tolerance requires you to install an SCPS license in your Steelhead appliance.
- SCPS rate pacing require you to install an SCPS license in your Steelhead appliance and configure MX-TCP through advanced QoS.

For details on MX-TCP and QoS, see [“MX-TCP” on page 93](#).

- SCPS compression.

Steelhead appliance transport settings are communicated among peers through the out-of-band connection between Steelhead appliance peers.

Note: All RiOS TCP stacks support selective acknowledgments (SACK) for efficient retransmission of packets.

Bandwidth Estimation

The bandwidth estimation transport setting uses an intelligent bandwidth estimation algorithm along with a modified slow-start algorithm to optimize performance in long lossy networks. These networks typically include satellite and other wireless environments, such as cellular networks, longer microwave, or Wi-Max networks. Bandwidth estimation is a sender-side modification of TCP and is compatible with the other TCP stacks in RiOS. The intelligent bandwidth estimation is based on analysis of both ACKs and latency measurements. The modified slow-start mechanism enables a flow to ramp up faster in high latency environments than traditional TCP. The intelligent bandwidth estimation algorithm allows it to learn effective rates for use during modified slow start, and also to differentiate BER loss from congestion-derived loss and deal with them accordingly. Bandwidth estimation has good fairness and friendliness qualities toward other traffic along the path.

Configuring Error Recovery

To handle satellite transmission errors and intermittent connectivity, RiOS v6.5 or later includes the lossy link-error recovery mechanism. In a lossy environment, you can enable error recovery to modify the congestion avoidance algorithm of the underlying TCP stack. This causes the underlying RiOS TCP stack congestion avoidance algorithm to be less responsive to retransmissions: for example, duplicate ACKs. This can be quite effective in situations with BER loss. However, you might not want it in situations where loss is congestion-based. By making TCP less responsive to loss, you might cause congestion to worsen. Due to this trade-off, use caution when you enable error recovery, particularly in situations with coexisting traffic along a path. Riverbed recommends that you do not enable error recovery on terrestrial channels.

You can enable error recovery only from the CLI. Use the **tcp err-recovery loss-recovery mode always** command on the client-side (remote) Steelhead appliance.

To configure satellite transmission error recovery

- Connect to the client-side (remote) Steelhead appliance and enter the following command:

```
tcp err-recovery loss-recovery mode always
restart
```

This enables lossy link error recovery on all traffic sent by this remote Steelhead appliance. This configuration is communicated to a peer Steelhead appliance so that the peer Steelhead appliance can use error recovery when it sends traffic to the remote Steelhead appliance. By default, error recovery is disabled.

To disable lossy link error recovery, use **tcp err-recovery loss-recovery mode disable**.

SCPS Per Connection

The SCPS per connection transport setting uses a modified slow-start algorithm and a modified congestion-avoidance approach. This enables SCPS per connection to ramp up flows faster in high-latency environments, and handle lossy scenarios, while remaining reasonably fair and friendly to other traffic. SCPS per connection does a very good job of efficiently filling up satellite links of all sizes. SCPS per connection is a high performance option for satellite networks.

SCPS Error Tolerance

The SCPS error tolerance transport setting uses a modified slow-start algorithm and a modified congestion avoidance approach. SCPS error tolerance requires many more retransmitted packets to trigger the congestion avoidance algorithm than the SCPS per connection. The assumption is that the environment in which you use SCPS error tolerance has a high BER and most retransmissions are due to poor signal quality instead of congestion. This enables SCPS error tolerance to efficiently maximize performance in high loss environments, without incurring the additional per-packet overhead of a FEC algorithm at the transport layer. SCPS error tolerance is a high performance option for lossy satellite networks.

Do not use SCPS error tolerance in clean networks, because it can be quite aggressive and compete unfairly with other traffic.

SCPS Rate Pacing

The SCPS rate pacing setting uses a modified slow start algorithm to intelligently switch to congestion avoidance without incurring the penalty of the first loss to exit TCP slow start. Additionally, the SCPS rate pacing mechanism maintains a steady data rate in congestion avoidance while efficiently adapting to congestion in a shared network.

This is a marked improvement over using SCPS per connection, SCPS error tolerant, or MX-TCP. SCPS per connection and error tolerant switch from slow start to congestion avoidance on the first loss event. MX-TCP does not adapt to congestion in a shared network and could cause congestion collapse in which senders continually retransmit data. The combination of an efficient mechanism to switch from slow start without incurring a loss event and the use of a steady data rate in the congestion avoidance phase enables the Steelhead appliance to fill a satellite link while avoiding some loss from buffer overruns and buffer delay for latency-sensitive TCP traffic.

Rate pacing is an ideal setting for many networks. The combination of a tuned TCP stack for satellite environments and MX-TCP to control the rate eases the burden of calculating buffer sizes and making adjustments across the infrastructure.

Rate pacing requires you to configure MX-TCP and as a result does not support IPv6.

For details on how to configure rate pacing, see [“Configuring Rate Pacing” on page 311](#).

SCPS Single-Ended Rules

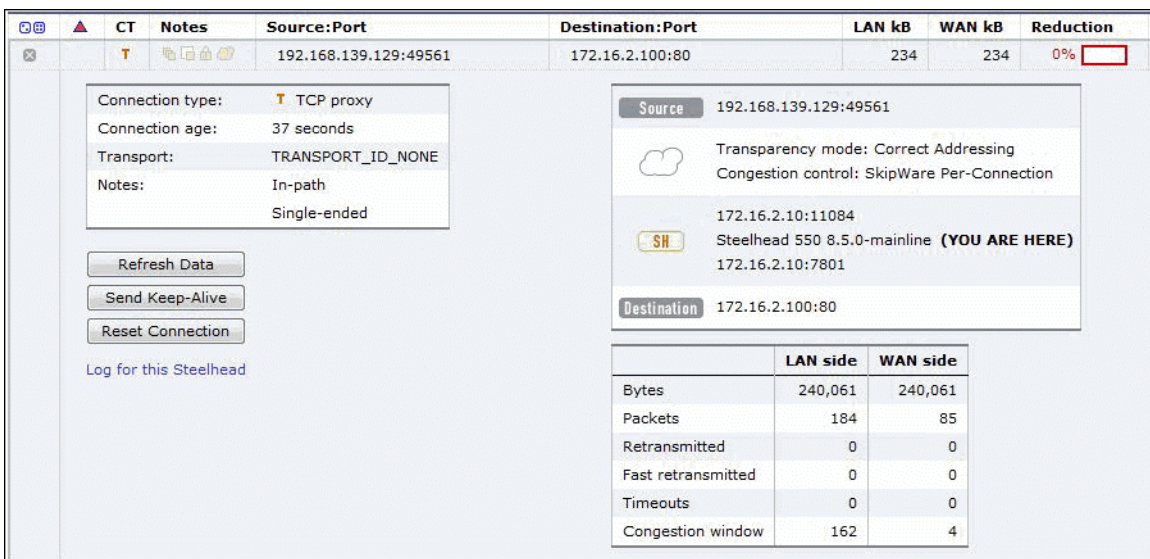
SCPS single-ended rules provide an option to use a TCP stack that is tuned for satellite environments when communicating with a third-party WAN optimizer or TCP-PEP. SCPS single-ended rules is used most commonly for SCPS integration.

In RiOS v8.5 or later, SCPS single-ended rules are enhanced to provide additional options suited for more complex environments. One key addition is the ability to select a transport option (TCP stack) on a per-rule basis, allowing you to use the most appropriate TCP stack for a given environment. You can also enable the rate pacing functionality on a per-rule basis to further adjust the transport optimization to the environment. You can support IPv6 hosts. The default rule is to pass through connections for all IPv4 and IPv6 hosts.

You can use SCPS single-ended rules to take advantage of a single-ended proxy. This functionality can help where there is no WAN optimizer or TCP-PEP on the other side of the connection. Because this feature is a sender-side modification, it provides the most benefit when used at the side of the connection sending the majority of the data.

Figure 13-1 shows an example of a single-ended proxy connection. In this example, the server-side Steelhead appliance uses a single-ended rule to proxy the connection using SCPS per connection as the TCP stack in conjunction with rate pacing.

Figure 13-1. SCPS Single-Ended Rule as a Single-Ended Proxy



For details on configuring single-ended rules, see [“Configuring Single-Ended Rules”](#) on page 313.

SCPS Compression

SCPS compression is available in RiOS v8.5 or later. SCPS compression is designed for scenarios in which the Steelhead appliance is interoperating with a third-party WAN optimizer or TCP-PEP using SCPS single-ended rules. When you enable SCPS compression, the Steelhead appliance negotiates SCPS compression functionality with the third-party device. Compression is performed across the TCP header and payload. The TCP header is compressed according to the SCPS-TP standard, and the payload uses LZ-based compression on a packet-by-packet basis. SCPS-compressed IP packets have a next-protocol value of 105, which can require changes to intervening security devices.

Configuring Automatic Detect TCP Optimization

Automatic detect TCP optimization is in RiOS v7.0 or later. Automatic detect TCP optimization enables a Steelhead appliance to detect the transport setting (in other words, the TCP stack) advertised by a peer Steelhead appliance and implement the respective transport setting for the applicable TCP flows to the peer Steelhead appliance. This is useful in complex topologies where several different types of networks terminate into a hub or server-side Steelhead appliance.

Use automatic detect TCP optimization on the hub Steelhead appliance to enable an organization that uses the best transport optimization for the respective network segment. You enable automatic detect TCP optimization on remote Steelhead appliances. Automatic detect TCP optimization is advertised to a peer via the out-of-band connection between the appliances. The transport setting is communicated by the client-side or server-side Steelhead appliance. However, at least one Steelhead appliance must support automatic detect TCP optimization; otherwise each Steelhead appliance uses its defined transport setting.

Use automatic detect TCP optimization when you have lower-speed terrestrial connected remote sites, high-speed data centers, and satellite sites all terminating into a Steelhead appliance at an aggregation point: for example, a data center. When you enable automatic detect TCP optimization on the aggregation point Steelhead appliance, and the desired transport settings on the remote Steelhead appliances, each network segment uses the appropriate transport setting.

To configure automatic detect TCP optimization

- Connect to the CLI and enter the following commands:

```
tcp cong-ctrl mode auto
write memory
```

You can also choose Configure > Optimization > Transport Settings, select Auto-Detect TCP Optimization, and click **Apply** and **Save**.

Integrating the Steelhead Appliance with Existing Satellite Modem TCP Acceleration

In some networks, the satellite modems might have built-in TCP acceleration and possibly even LZ compression. Typically, these solutions do not perform well as advanced WAN optimizers such as a Steelhead appliance. However, when you deploy a Steelhead appliance into an environment with TCP acceleration in the satellite modems, the challenge is learning if your modems have TCP acceleration enabled, and then determining the most appropriate integration method.

You might not be aware if the modems you have are providing acceleration service until you have deployed the Steelhead appliances, and auto-discovery does not work due to the satellite modem stripping the RiOS TCP option entries used for auto-discovery.

For details on how to analyze for stripped probes, see [“Verification and Troubleshooting” on page 315](#).

You can choose from several options to integrate Steelhead appliances with satellite modems conducting TCP acceleration. The options include:

- Disable TCP acceleration in the satellite modems.
- Enable TCP option forwarding in the satellite modem for the appropriate TCP options:
 - SCPS = TCP option 20
 - Correct Addressing = TCP option 76
 - Port Transparency = TCP option 76 and 78
 - Full Transparency = TCP option 76 and 78
- Use fixed-target rules in the Steelhead appliances instead of auto-discovery. For more details on fixed-target in-path rules, see the *Steelhead Appliance Management Console User’s Guide*.

For details on how to disable TCP acceleration or configure TCP option forwarding, contact the satellite modem vendor technical support.

Licensing SCPS on a Steelhead Appliance

To determine if your Steelhead appliance has a valid SCPS license, choose Configure > Maintenance > Licenses and look in the description column for SCPS. You can also use the **show licenses** command in the CLI. The format of the actual license is LK1-SH55SCPS-XXXX-XXXX-1-XXXX-XXXX-XXXX.

When you order an SCPS license, it is delivered by email from riverbed-license@riverbed.com. You need a license for the data center and branch office Steelhead appliance.

To install an SCPS license

1. Choose **Configure > Maintenance > Licenses**.
2. Click **Add License**.
3. Paste the license into the text field.
4. Click **Apply**.

Or, connect to the CLI and enter the following commands:

```
enable
config t
license install <your license key>
wr me
restart
```

You must perform an optimization service restart to complete the installation of the SCPS license.

Configuring Satellite Optimization Features

This section contains the following topics:

- [“Configuring Transport Optimization” on page 307](#)
- [“Configuring Rate Pacing” on page 311](#)
- [“Configuring Single-Ended Connection Rule Table Settings” on page 311](#)
- [“Configuring Single-Ended Rules” on page 313](#)

Configuring Transport Optimization

To properly configure transport settings for the target environment, you must understand its characteristics. This section describes how to configure, monitor, and troubleshoot the transport settings in RiOS v7.0 or later.

To capture your performance characteristics

1. Connect to the Steelhead appliance CLI, using an account with administration rights.
2. Enter the enable mode and then configure terminal mode:

```
en
conf t
```

3. If your environment does not support path MTU discovery, use the ping command to measure the maximum transmission unit (MTU) by pinging a remote host.

Start with a full-size packet and decrease the size of the packet in small increments until the ping successfully reaches the target host.

Write the MTU here: _____

Write the round trip time here: _____

If you are deploying your Steelhead appliance through WCCP, you might need to take into account the additional GRE overhead of WCCP. The following shows an example ping command for measuring maximum packet size along a specified path.

```
ping -I inpath0_0 -s <Bytes> <target host>
```

4. Use the following command with a full-size packet for a count of 1000 or more packets.

```
ping -I inpath0_0 -c 1000 -s <your MTU> <target host>
```

Write the percentage of loss during this test here: _____

5. Configure the Steelhead appliance WAN buffers for the target network.

Using the data you have collected, calculate five times bandwidth delay project (BDP) for your satellite network using the following formula or table. For satellite networks that vary in capacity, use the maximum potential speed the network can achieve. If your satellite link speeds differ in either direction, you might have different size *send* and *receive* buffer sizes. If this is the case, your send buffer on the transmitting side should match your receive buffer on the receiving side.

Steelhead appliance WAN Buffers = ((Your RTT in milliseconds * 0.001) * (Your Circuit Speed in bps/8) * 5)

For example, ((600ms * 0.001) * (1,544,000bps/8) * 5) = 579,000 byte WAN buffers.

Use the following table as a quick reference to help estimate appropriate Steelhead appliance WAN buffers.

Link Speed (bps)	256,000	768,000	1,544,000	6,000,000	10,000,000	20,000,000	45,000,000
RTT (ms)							
600	96,000	288,00	579,000	2,250,000	3,750,000	7,500,000	16,875,000
700	112,000	336,00	675, 500	2,625,000	4,375,000	8,750,000	19,687,500
800	128,000	384,000	772,000	3,000,000	5,000,000	10,000,000	22,500,000
900	144,000	432,000	868,500	3,375,000	5,625,000	11,250,000	25,312,500
1000	160,000	480,000	965,000	3,750,000	6,250,000	12,500,000	28,125,000
1100	176,000	528,000	1,061,500	4,125,000	6,875,000	13,750,000	30,937,500
1200	192,000	576,000	1,158,000	4,500,000	7,500,000	15,000,000	33,750,000
1300	208,000	624,000	1,254,500	4,875,000	8,125,000	16,250,000	36,562,500
1400	224,000	672,000	1,351,000	5,250,000	8,750,000	17,500,000	39,375,000
1500	240,000	720,000	1,447,500	5,625,000	9,357,000	18,750,000	42,187,500
1600	256,000	768,000	1,544,000	6,000,000	10,000,000	20,000,000	45,000,000
1700	272,000	816,000	1,640,500	6,375,000	10,625,000	21,250,000	47,812,500
1800	288,000	864,000	1,737,000	6,750,000	11,250,000	22,500,000	50,625,000
1900	304,000	912,000	1,833,500	7,125,000	11,875,000	23,750,000	53,437,500
2000	320,000	960,000	1,930,000	7,500,000	12,500,000	25,000,000	56,250,000

Write the WAN buffer size, in bytes, here: _____

6. Configure the Steelhead appliance LAN buffers for the target network to be 1/100th the WAN buffer, but no less than 50,000 bytes.

Write the LAN buffer size, in bytes, here: _____

To configure transport settings

1. Configure all the Steelhead appliance WAN buffers with the following commands:

```
protocol connection wan send def-buf-size <your buffer size>
protocol connection wan receive def-buf-size <your buffer size>
protocol connection lan send buf-size <your buffer size>
protocol connection lan receive buf-size <your buffer size>
```

Or, choose Configure > Optimization > Transport Optimization, select WAN and LAN receive and send buffers, and click **Apply**.

2. Configure your remote Steelhead appliances with the desired transport options from the commands in the following table.

Transport Optimization Option	CLI Command
Enable BW estimation	<code>tcp-cong-ctrl mode bw-est</code>
Enable error recovery	<code>tcp-err-recovery loss-recovery mode always</code>
Disable error recovery	<code>tcp-err-recovery loss-recovery mode disable</code>
Enable SCPS per connection	<code>tcp cong-ctrl mode per-conn-tcp</code>
Enable SCPS error tolerance	<code>tcp cong-ctrl mode err-tol-tcp</code>
Set back to default TCP	<code>tcp cong-ctrl mode default</code>

Or, choose Configure > Optimization > Transport Settings, select the appropriate radio button, and click **Apply** (Figure 13-2).

Figure 13-2. Transport Settings Page

Configure > Optimization > Transport Settings ?

TCP Optimization

Congestion Control Algorithm

☐ Auto-Detect
☒ Standard (RFC-Compliant)
☐ HighSpeed
☐ Bandwidth Estimation
☐ SkipWare Per-Connection
☐ SkipWare Error-Tolerant

☐ Enable Rate Pacing

Buffer Settings

LAN Send Buffer Size: bytes
 LAN Receive Buffer Size: bytes
 WAN Default Send Buffer Size: bytes
 WAN Default Receive Buffer Size: bytes

Single-Ended Connections

☐ Enable Single-Ended Connection Rules Table
☐ Enable SkipWare Legacy Compression

Apply

Single-Ended Connection Rules:

+ Add New Rule - Remove Selected Rules ⬆⬆ Move Selected Rules...

<input type="checkbox"/>	Rule	Source	Destination	VLAN	Traffic
<input type="checkbox"/>	1	All-IP	All-IP:Interactive	All	Passthrough
<input type="checkbox"/>	2	All-IP	All-IP:RBT-Proto	All	Passthrough
<input type="checkbox"/>	default	All-IP	All-IP:All	All	Passthrough

3. If you have a mixed environment, configure your hub Steelhead appliance to use automatic detect TCP optimization to reflect the various transport optimization mechanisms of your various remote site Steelhead appliances.

You can also hard code your hub Steelhead appliance to the desired setting.

4. Restart the optimization service, either with the Management Console or the CLI.

Riverbed recommends that you test a few different transport settings at different remote sites and determine which settings work best for your environment.

For details on automatic detect TCP, see [“Configuring Automatic Detect TCP Optimization” on page 305](#). For details on gathering performance characteristics and configuring transport settings, see the *Riverbed Command-Line Interface Reference Manual* and the *Steelhead Appliance Management Console User’s Guide*.

Configuring Rate Pacing

The following steps are required for rate pacing to function:

1. After you chose the transport option, select the Enable Rate Pacing check box in the Steelhead Management Console. You can also use the CLI command **tcp rate-cap enable**.
2. Configure MX-TCP under Advanced QoS.

For more information about rate pacing, see [“SCPS Rate Pacing” on page 304](#). For more details on MX-TCP and QoS, see [“MX-TCP” on page 93](#).

The relationship between the overall link rate and MX-TCP rate dictates how the rate pacing mechanism operates. Rate pacing exits TCP slow start at the MX-TCP rate if the MX-TCP rate is less than the link rate. If you configure rate pacing in this way, it avoids the first loss on exiting slow start and uses MX-TCP as a scheduler for sending data while still adapting to congestion on a shared network in the congestion avoidance phase.

Alternatively, if the MX-TCP rate is greater than the link rate, then rate pacing exits at the MX-TCP rate. This exit rate can incur a loss on exiting slow start, or packets are buffered in the bottleneck queue. The sending rate during congestion avoidance is based on a calculation between the rate the transport option (TCP stack) determines and the MX-TCP rate. Over time, the rate pacing mechanism continually probes for the higher MX-TCP rate.

In summary, the relationship works as follows:

- **Link rate greater than MX-TCP rate**—Exit slow start at MX-TCP rate and maintain MX-TCP rate in congestion avoidance.
- **Link rate is greater than 50% of the MX-TCP rate but less than the MX-TCP rate**—Exit slow start at MX-TCP rate and use the congestion avoidance rate determined by the underlying TCP stack selected as the transport option.
- **Link rate less than 50% of the MX-TCP rate or MX-TCP not enabled**—Use the underlying transport option for exiting slow start and the congestion avoidance algorithm.

Because a hub site can be connected to multiple satellite networks and remote sites can use a variety of TCP stacks, it is appropriate for you to use automatic detect on the hub site for rate pacing. You can set up MX-TCP on a site-by-site basis to refine the data rate for each remote. MX-TCP follows the QoS configuration for matching on a site and rule.

The following is an example configuration for the hub site for two remote sites using rate pacing with different bandwidths:

- Site 1 has subnet 172.16.1.0/24 and a link rate of 2 Mbps
- Site 2 has subnet 172.16.2.0/24 and a link rate of 8 Mbps

Use the following CLI commands on the hub-site Steelhead appliance:

```
tcp cong-ctrl mode auto
tcp rate-cap enable
```

Configuring Single-Ended Connection Rule Table Settings

Use the single-ended connection rule table to manage which flows are optimized or passed through for SCPS optimization. Configuration of the single-ended optimization rule table is similar to the in-path rules, but you must enable the single-ended connection rule table to apply the rules.

To enable the single-ended connection rule table

- Connect to the CLI and enter the following command:

```
tcp sat-opt scps scps-table enable
```

You must have RiOS v8.5 or later to enable the single-ended connection rule table and SCPS compression with third-party WAN optimizers or TCP-PEPs.

To enable the single-ended connection rule table and SCPS compression with third-party WAN optimizers or TCP-PEPs

- Connect to the CLI and enter the following command:

```
tcp sat-opt scps scps-table enable
tcp sat-opt scps legacy-comp enable
```

You can also complete the following procedure from the Steelhead Management Console Configure > Optimization > Transport Settings page.

Figure 13-3. Transport Settings Page with Single-Ended Connection Rule and SCPS Compression

Configure > Optimization > Transport Settings ?

TCP Optimization

Congestion Control Algorithm

- ☐ Auto-Detect
- ☐ Standard (RFC-Compliant)
- ☐ HighSpeed
- ☐ Bandwidth Estimation
- ☒ SkipWare Per-Connection
- ☐ SkipWare Error-Tolerant

☒ Enable Rate Pacing

Buffer Settings

LAN Send Buffer Size: bytes

LAN Receive Buffer Size: bytes

WAN Default Send Buffer Size: bytes

WAN Default Receive Buffer Size: bytes

Single-Ended Connections

☒ Enable Single-Ended Connection Rules Table

☒ Enable SkipWare Legacy Compression

Enabling the SCPS single-ended connection rule table or SCPS compression requires a service restart.

To see the current rules in the table, use the **show tcp sat-opt scps rules** command. Following is an example single-ended connection rule table:

```
ssh (config) # show tcp sat-opt scps rules
Rule S P VLAN Source Addr      Dest Addr      Port
-----
  1 Y Y all  all              all            Interactive
  2 Y Y all  all              all            RBT-Proto
def Y Y all  all              all            all
```

```
(S) SCPS setting:          Y=Allow SCPS
                        N=SCPS Bypass
(P) Allow only SCPS peering: Y=Enabled
                        N=Disabled
```

Rules are matched from top to bottom. A flow matching a rule stops at the first rule it matches and applies one of the SCPS modes: pass-through or enable. To pass through all flows for SCPS optimization, add a rule at the start of the table to match all sources, all destinations, all destination ports, and all VLANs.

To create a pass through all flows rule

- Connect to the CLI and enter the following command:

```
tcp sat-opt scps rule srcaddr all dstaddr all dstport "all" allow-scps disable scps-peer-only
disable rulenum start
```

Figure 13-6 shows an example of a pass-through rule in the Management Console.

Configuring Single-Ended Rules

The following are procedures for configuring single-ended rules. Configuration of single-ended rules in RiOS v8.5 or later differs from configuration in RiOS v7.0 and v8.0. There are additional options available in single-ended rules: for example, using a single-ended proxy, enabling SCPS discovery or third-party integration and using rate pacing.

Figure 13-4 shows an example of adding a single-ended rule configured in the Steelhead Management Console using SCPS per connection as the TCP stack and rate pacing enabled. You can use this configuration for when you interpolate with a third-party WAN optimizer or TCP-PEP and your network could benefit from using rate pacing with SCPS per connection as the transport option. Rate pacing requires that you configure MX-TCP with advanced QoS and MX-TCP only supports IPv4 traffic.

Figure 13-4. Single-Ended Rule with SCPS Per Connection and Rate Pacing

Single-Ended Connection Rules:

▼ Add New Rule — Remove Selected Rules ⬆⬆ Move Selected Rules...

Position: End

Source Subnet: All-IP

Destination Subnet: All-IP Port or [Port Label](#): all

VLAN Tag ID: all

Traffic

For "passthrough" (no optimization) uncheck both "SCPS Discover" and "TCP Proxy."

Status: Optimized

☒ SCPS Discover

☐ TCP Proxy

TCP Optimization

Congestion Control Algorithm: SkipWare Per-Connection

☒ Enable Rate Pacing

Add

For more details on single-ended rules, see [“SCPS Single-Ended Rules” on page 304](#). For details on configuring single-ended before RiOS v8.5, see earlier versions of the *Steelhead Appliance Deployment Guide* and *Riverbed Deployment Guide* on the Riverbed Support site at <https://support.riverbed.com>.

To edit a single-ended connection rule

1. Choose Configure > Optimization > Transport Settings page.
2. Click the magnifying glass by the rule that you want to edit.

Figure 13-5. Edit Single-Ended Connection Rules

Single-Ended Connection Rules:

+ Add New Rule - Remove Selected Rules ↑↓ Move Selected Rules...

<input type="checkbox"/>	Rule	Source	Destination	VLAN	Traffic	SCPS Discover
<input type="checkbox"/>	1	All-IP	All-IP:Interactive	All	Passthrough	--
<input type="checkbox"/>	2	All-IP	All-IP:RBT-Proto	All	Passthrough	--
<input checked="" type="checkbox"/>	3	All-IP	All-IP:All	All	Optimized	Enabled

Source Subnet:

Destination Subnet:

VLAN Tag ID:

Port or [Port Label](#):

Traffic

For "passthrough" (no optimization) uncheck both "SCPS Discover" and "TCP Proxy."

Status: Optimized

☒ SCPS Discover

☐ TCP Proxy

TCP Optimization

Congestion Control Algorithm:

☒ Enable Rate Pacing

default	All-IP	All-IP:All	All	Passthrough	--

To add a single-ended connection rule

1. Choose Configure > Optimization > Transport Settings page.
2. Select Add New Rule.
3. Populate the appropriate fields and settings.
4. Click **Add**.

The changes take place immediately to all new flows.

Figure 13-6 shows how to configure a pass-through rule for all traffic. Clear the SCPS Discover and TCP Proxy check boxes.

Figure 13-6. Configure A Pass-Through Rule for All Traffic

Single-Ended Connection Rules:

▼ Add New Rule — Remove Selected Rules ⇅ Move Selected Rules...

Position: End ▼

Source Subnet: All-IP

Destination Subnet: All-IP Port or Port Label: all

VLAN Tag ID: all

Traffic

For "passthrough" (no optimization) uncheck both "SCPS Discover" and "TCP Proxy."

Status: Passthrough

☐ SCPS Discover

☐ TCP Proxy

TCP Optimization

Congestion Control Algorithm: SkipWare Per-Connection ▼

☐ Enable Rate Pacing

Add

Figure 13-7, Rule 1, shows an example of a single-ended optimization pass-through rule for all traffic initiated from the client-side Steelhead appliance.

Figure 13-7. Single-Ended Optimization Pass-Through Rule

Single-Ended Connection Rules:						
+ Add New Rule — Remove Selected Rules ⇅ Move Selected Rules...						
<input type="checkbox"/>	Rule	Source	Destination	VLAN	Traffic	SCPS Di
<input type="checkbox"/>	1	All-IP	All-IP:All	All	Passthrough	---
<input type="checkbox"/>	2	All-IP	All-IP:Interactive	All	Passthrough	---
<input type="checkbox"/>	3	All-IP	All-IP:RBT-Proto	All	Passthrough	---
	default	All-IP	All-IP:All	All	Passthrough	---

The Management Console passes through only locally initiated sessions through the LAN interface. Inbound SCPS sessions (SYNs with SCPS negotiation headers) arriving at the WAN interface are terminated. To bypass these inbound SCPS sessions, use the CLI. To pass through inbound SCPS sessions in the single-ended connection table, use the syntax option **scps-peer-only disable**.

Verification and Troubleshooting

This section describes common satellite deployment problems and solutions. This section includes the following topics:

- [“Analyzing Connection Optimization Information” on page 316](#)
- [“Analyzing Packets for Discovery Probe Stripping” on page 319](#)

- [“Understanding the Health of the Satellite Signal” on page 321](#)
- [“Potential Performance Impact of Loss at the Start of Flow” on page 322](#)
- [“Variance in SCPS Performance” on page 322](#)

Analyzing Connection Optimization Information

After you configure the Steelhead appliance transport settings, you can verify if the solution is working as expected. You can determine which transport optimization method a connection is using on the Current Connections page in the Management Console and the CLI. This section includes the following:

- [“Using the Steelhead Management Console to Investigate Connection Details” on page 316](#)
- [“Using the Riverbed Command-Line Interface to Investigate Connection Details” on page 318](#)

Using the Steelhead Management Console to Investigate Connection Details

The Current Connections page in the Management Console provides extensive details on flows observed by the Steelhead appliance. You can efficiently determine various details on flows using this report. To see the details for a flow, including transport settings, click the magnifying glass of a specific connection.

Figure 13-8. Current Connections

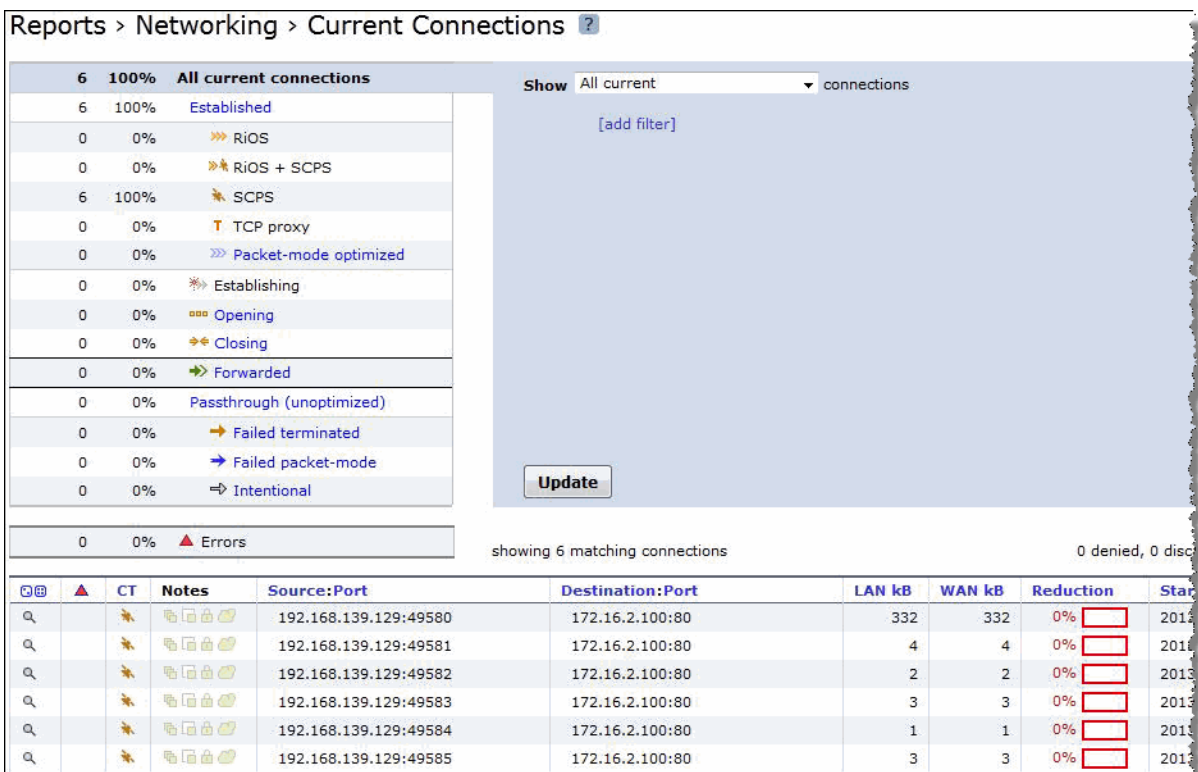
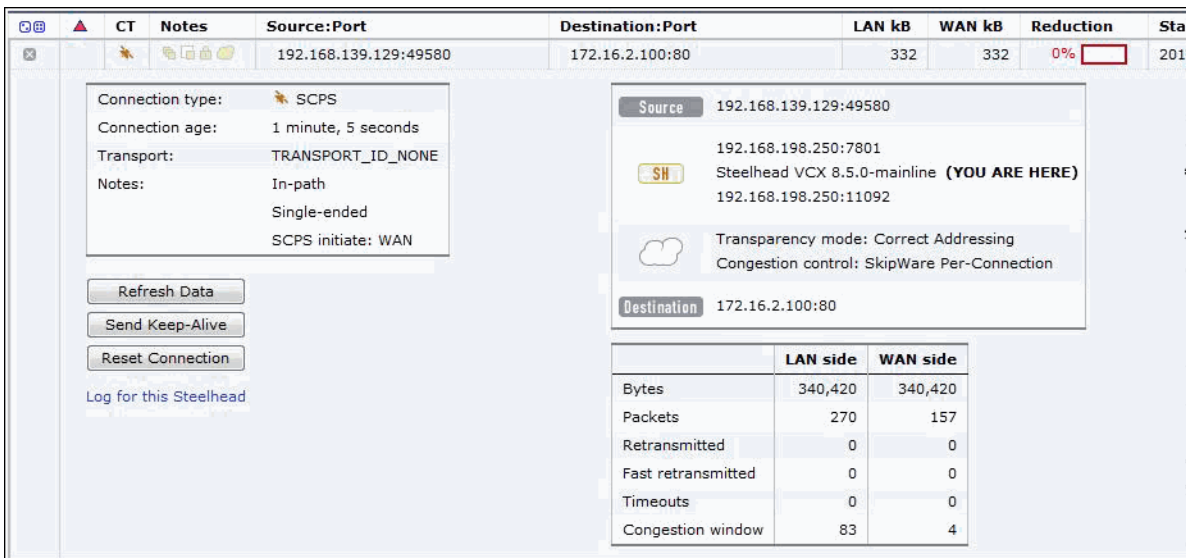


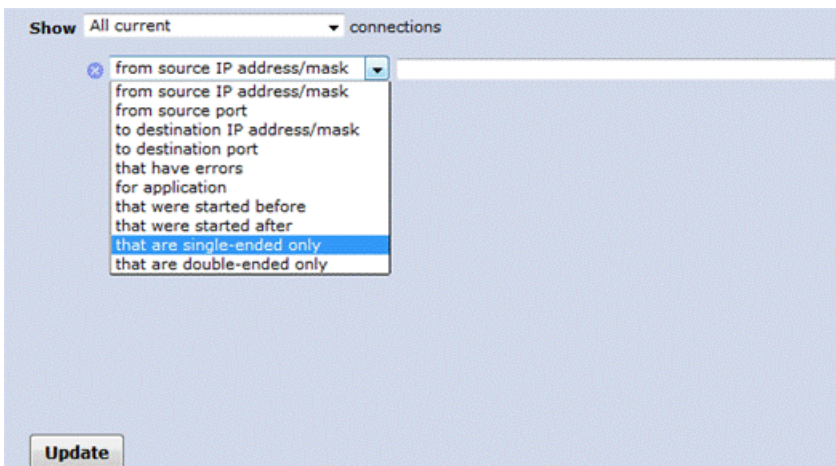
Figure 13-9 shows the details of a specific connection. This flow is single-ended optimized, which confirms it is using SCPS. The SCPS Initiate is set to WAN, which indicates that this Steelhead appliance is on the client side of this session. If the SCPS Terminate is WAN, then this Steelhead appliance is the server-side for the flow. The SCPS Initiate and Terminate cannot both read WAN, because a TCP flow can be initiated only in a single direction. The WAN Congestion Control indicates the transport setting, which in this case is per-connection.

Figure 13-9. Connection Information



If the current connections report has a lot of flows, you can filter your view. To see only single-ended optimization flows, show All current connections, add a filter, and in the drop-down list select *that are single-ended only* as shown in Figure 13-10.

Figure 13-10. Filtering Flow View



Using the Riverbed Command-Line Interface to Investigate Connection Details

The CLI provides extensive details on flows observed by the Steelhead appliance. The **show connections** command provides a summary list of the connections flowing through a Steelhead appliance. You can use this command with SCPS to quickly see which flows are optimized (O), single-ended optimized (S), and which flows are pass-through (P, PI, PU). Singled-ended optimized flows are included in the established optimized flow total of the **show connections** command. If you want more detail, use the **show connections optimized full** command. See the following examples for details.

```
SH # show connections
T Source                Destination            App    Rdn Since
-----
SO 192.168.139.129:49588 172.16.2.100:80      TCP    0% 2013/07/01 05:26:03
SO 192.168.139.129:49589 172.16.2.100:80      TCP    0% 2013/07/01 05:26:03
SO 192.168.139.129:49590 172.16.2.100:80      TCP    0% 2013/07/01 05:26:03
SO 192.168.139.129:49591 172.16.2.100:80      TCP    0% 2013/07/01 05:26:05
SO 192.168.139.129:49592 172.16.2.100:80      TCP    0% 2013/07/01 05:26:05
SO 192.168.139.129:49593 172.16.2.100:80      TCP    0% 2013/07/01 05:26:05
SO 192.168.139.129:49594 172.16.2.100:80      TCP    0% 2013/07/01 05:26:05
SO 192.168.139.129:49595 172.16.2.100:80      TCP    0% 2013/07/01 05:26:05
SO 192.168.139.129:49596 172.16.2.100:80      TCP    0% 2013/07/01 05:26:05
-----
                                     All    V4    V6
-----
Established Optimized:                9      9      0

  RiOS Only (O):                      0      0      0
  SCPS Only (SO):                     9      9      0
  RiOS+SCPS (RS):                     0      0      0
  TCP Proxy (TP):                     0      0      0

Half-opened optimized (H):            0      0      0
Half-closed optimized (C):            0      0      0
Establishing (E):                     0      0      0

Pass Through:                         0      0      0

  Passthrough Intentional (PI):        0      0      0
  Passthrough Unintentional (PU):      0      0      0

Forwarded (F):                        0      0      0
Discarded (not shown):                 0
Denied (not shown):                    0
-----
Total:                                9      9      0
```

For more detail, use the **show connection** command options. The syntax requires very specific inputs, and it must be executed while the flow is established through the Steelhead appliance:

```
show connection srcip <IP ADDR> srcport <PORT> dstip <IPADDR> dstport <PORT>
```

You can look at the IP address and port requirements in the show connections flow table. An example of this command follows. The TCP congestion control mechanism is listed in the middle after WAN visibility mode:

```
SH # show connection srcip 192.168.139.129 srcport 49588 dstip 172.16.2.100 dstport 80
Connection not found.
SH # show connection srcip 192.168.139.129 srcport 49597 dstip 172.16.2.100 dstport 80
Type:                               Single-ended optimized
Optimization Policy:                 None
Source:                             192.168.139.129:49597
Destination:                         172.16.2.100:80

Application:                         TCP
```



```

Reduction:                0%
Since:                    2013/07/01 05:27:19

Cloud Acceleration State:  None

Source Side Statistics:
TCP Congestion Algorithm:  Skipware Per Connection
  Bytes:                  328301
  Packets:                273
  Retransmitted:          0
  Fast Retransmitted:     0
  Timeouts:               0
  Congestion Window:      235

Destination Side Statistics:
TCP Congestion Algorithm:  New Reno
  Bytes:                  328301
  Packets:                105
  Retransmitted:          0
  Fast Retransmitted:     0
  Timeouts:               0
  Congestion Window:      4

```

In most situations, it is easier to use the Current Connections page rather than the CLI for flow investigation. For details, see [“Using the Steelhead Management Console to Investigate Connection Details” on page 316.](#)

Analyzing Packets for Discovery Probe Stripping

RiOS auto-discovery and SCPS both rely on TCP options to function properly. Some network devices might strip the TCP options and negatively impact discovery or SCPS. In satellite environments, the satellite modems can have TCP acceleration enabled, which might strip TCP options and prevent the Steelhead appliances from automatically discovering one another.

This section describes how to troubleshoot this issue. You can confirm that TCP options are being stripped by capturing the SYN and SYN/ACK packets on the WAN interface of the server-side Steelhead appliance and looking for TCP options decimal 76 and/or 78. If you are using SCPS, also look for TCP option decimal 20.

On the server-side Steelhead appliance, you can use the following commands to capture only SYN and SYN/ACK packets for the wan0_0 interface:

```

enable
tcpdump -i wan0_0 -s 150 -w myfilename.cap 'tcp[13] & 2 = 2'

```

Note: Press CTRL+C to stop the packet capture from the CLI.

You can also execute and stop the capture in the Management Console Reports > Diagnostics > TCP Dumps page. From this page, you can download the capture file. If you are running RiOS v7.0 or later, you can use Pilot Enterprise to remotely start, stop, and analyze packet captures.

After you have downloaded the capture file, open it with a packet analyzer.

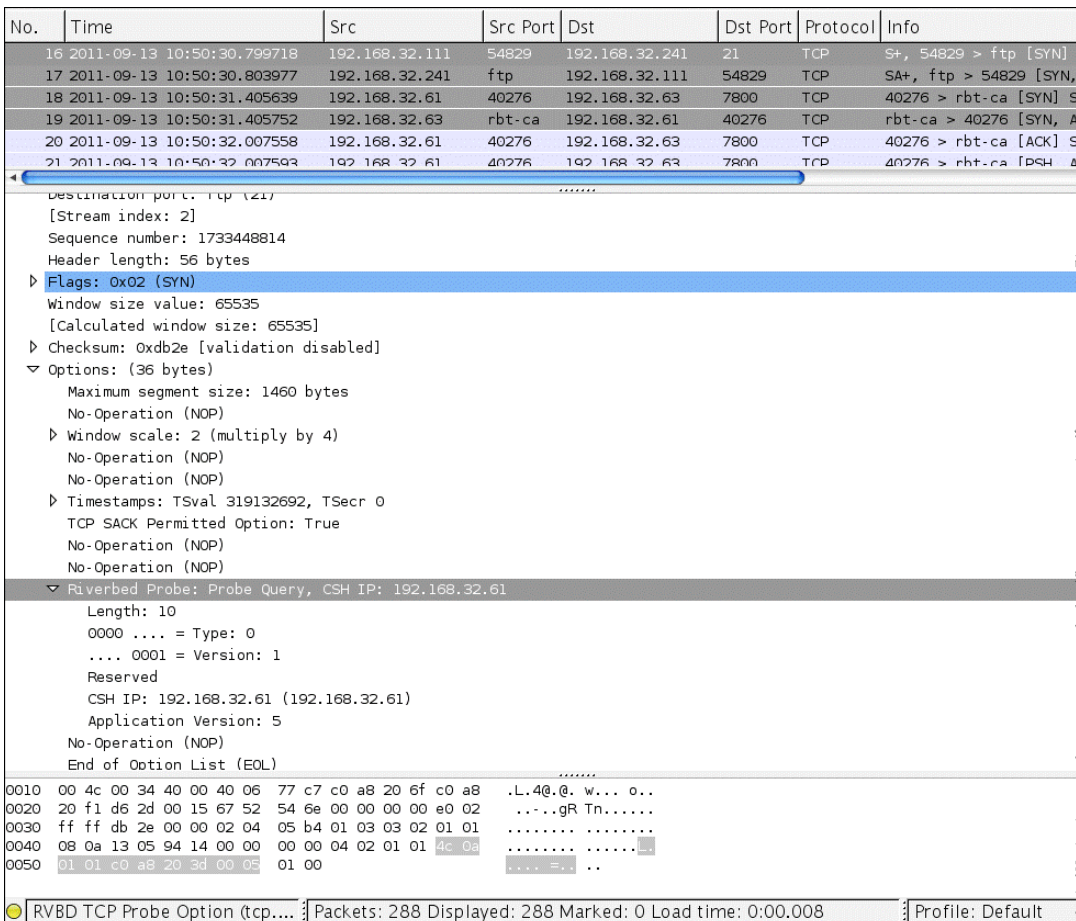
If you are using Wireshark 1.6.1 or later to analyze packets, the information row in the Pack List pane begins with S+ or SA+ if a RiOS auto-discovery probe is present in the TCP option field of a SYN or SYN/ACK, respectively. If you have many packets, use the Wireshark display filter `tcp.options.rvbd.probe==1` to display packets with RiOS discovery probes in the TCP option field.

To filter just for SCPS TCP options, use the display filter `tcp.options.scps==1`. Remember that SCPS TCP options are applied to only SYN or SYN/ACK packets. To filter for both RiOS and SCPS discovery probes, use the Wireshark display filter `tcp.options.rvbd.probe==1 || tcp.options.scps==1`. If you do not see any packets with RiOS or SCPS discovery probes, you likely have satellite modems stripping the TCP options field due to TCP acceleration.

After you select the desired packet, inspect the TCP option field to confirm that if the appropriate discovery probes are present. If they are not present at the server-side Steelhead appliance, some device is stripping the probes: for example, a satellite modem or firewall.

Figure 13-11 shows that the SYN packet (#16) is highlighted in Wireshark. In the Packet List pane, the Information column starts with S+, which denotes that the packet has a RiOS discovery probe in the TCP option field. In the Packet Details pane, the TCP option entry from the Steelhead appliance is highlighted in gray, and the details of the probe are decoded. In the Packet Bytes pane, you can see that the actual bytes for the RiOS discovery probe are highlighted and begin with 4c (0x4c is the hexadecimal representation of decimal 76).

Figure 13-11. Packet Information in Wireshark



If you use Wireshark often to analyze Steelhead appliance performance, you can use color filters to differentiate traffic.

To create Wireshark color filters

1. In Wireshark, choose View > Coloring Rules.

The Coloring Rules dialog box opens.

2. Click **New**.
3. Create a filter name, enter the desired display filter, and set your desired colors.
4. Click **OK**.

You can move the new color rule up or down so that it matches traffic accordingly. Remember that the first coloring rule that is matched is applied to the packet, so the order of color rules is very important.

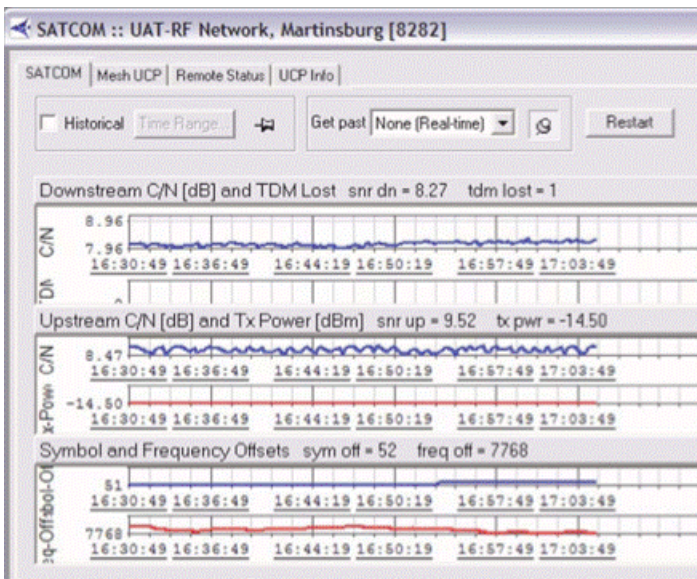
Understanding the Health of the Satellite Signal

Terms such as *signal-to-noise ratio*, *TDM loss*, and other satellite words and abbreviations might be foreign to you. To assist in troubleshooting, you should have a team of satellite experts in your NOC or your service provider's NOC/teleport. When you contact them, the primary questions you want to understand are:

- What is the utilization of the remote site's channel?
- What is the bit error rate for the specific remote site, and is it within an environment's comfort zone?

To analyze a problem, most satellite engineers have a management tool available to track performance. For example, you can use iDirect iMonitor to monitor the health of iDirect hub and remote equipment. Using iMonitor's SATCOM view, you can track performance on a satellite link for an individual remote on the upstream and downstream channels. [Figure 13-12](#) shows a graph from the SATCOM view in iDirect iMonitor.

Figure 13-12. SATCOM



Having your own network monitoring equipment to monitor TCP health, specifically TCP loss/retransmissions, provides an additional tool in the network infrastructure to monitor network health. Riverbed Cascade, Shark, and Pilot are capable of monitoring these metrics at various granularities.

Potential Performance Impact of Loss at the Start of Flow

TCP flows are most vulnerable to loss at the beginning of the flow. This is due to the initial TCP window size, which is very small at startup. When a TCP flow detects a lost packet in the first several turns, it can negatively impact the acceleration of the flow. Due to the latency in satellite networks, when this occurs, some TCP stacks take significantly longer to recover and accelerate the flow up to a reasonable speed.

When testing in labs, it is very important that you execute adequate flows against each test so that you capture a valid statistical sampling. This is critical because loss that coincidentally occurs at the start of flow negatively influences a single test for a certain vendor (Vendor A). Whereas the same test for Vendor B might not realize loss at the beginning of a flow, it might perform much better due to where the loss occurred, relative to the flow's life, and not specifically due to a more superior technology.

Variance in SCPS Performance

You might find that the TCP stacks of third-party SCSP solutions vary significantly. This can lead to different performance results when running the same transaction or test. When interoperating, you can find variance in performance between third-party devices, or variance depending on the direction data is transmitted. If you have questions or concerns about variance between SCPS solutions, it is best to engage all vendors in a joint discussion. Riverbed recommends that you have device configurations and packet captures from all devices to analyze during the discussion.

CHAPTER 14 VPN Routing and Forwarding

This chapter describes how to deploy Steelhead appliances in an MPLS/VRF environment using Not-So-VRF (NSV). NSV is a Riverbed network design option that leverages the Riverbed WDS solution by deploying Steelhead appliances in an existing MPLS deployment using virtual routing and forwarding (VRF).

This chapter includes the following sections:

- [“Overview of NSV with VRF Select” on page 323](#)
- [“Example NSV Network Deployment” on page 326](#)
- [“Configuring NSV” on page 328](#)

Overview of NSV with VRF Select

This section provides an overview of NSV. This section includes the following topics:

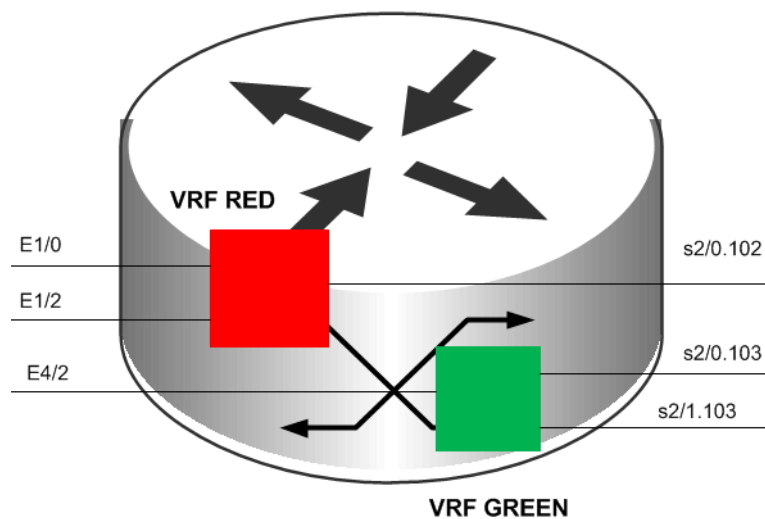
- [“Virtual Routing and Forwarding ” on page 324](#)
- [“NSV with VRF Select ” on page 325](#)
- [“IOS Requirements” on page 326](#)
- [“Prerequisites for NSV” on page 326](#)

Virtual Routing and Forwarding

Virtual routing and forwarding (VRF) is a technology used in computer networks that allows multiple instances of a routing table to coexist within the same router at the same time. VRF partitions a router by creating multiple routing tables and multiple forwarding instances. Because the routing instances are independent, you can use the same or overlapping IP addresses without conflict.

Note: The VRF table is also referred to as the *VPNv4 routing table*.

Figure 14-1. Partitioned Router Using Two Routing Tables

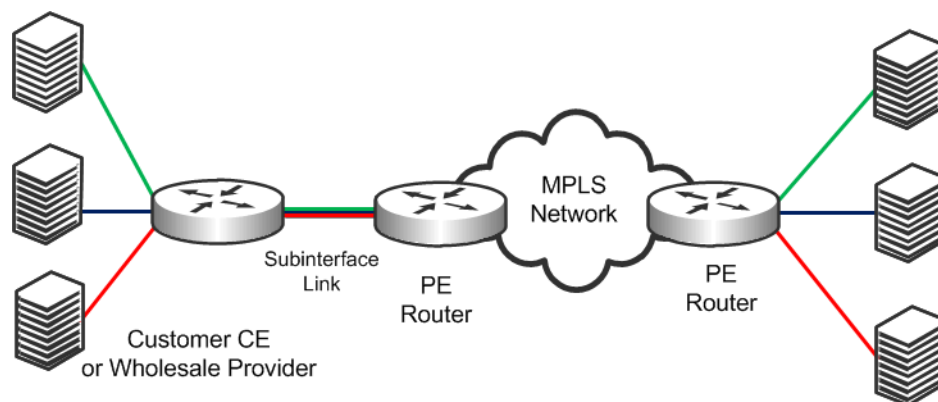


You can implement VRF in a network device by having distinct routing tables, one per VRF. Dedicated interfaces are bound to each VRF.

In [Figure 14-1](#), the red table can forward packets between interfaces E1/0, E1/2, and S2/0.102. The green table, on the other hand, forwards between interfaces E4/2, S2/0.103, and S2/1.103.

The simplest form of VRF implementation is VRF Lite, as shown in [Figure 14-2](#). VRF Lite uses VRFs without multiprotocol label switching (MPLS). In this implementation, each router within the network participates in the virtual routing environment in a peer-based fashion. This extends multiple VPNs from a provider edge (PE) device onto non-MPLS customer edge (CE) devices, which support multiple VRFs. It also replaces the requirement for separate, physical CE devices.

Figure 14-2. VRF Lite



NSV with VRF Select

NSV is a Riverbed network design option that leverages the Riverbed WDS solution by deploying Steelhead appliances in an existing MPLS deployment using VRF. Riverbed recommends using NSV in a MPLS/VRF environment to deploy Steelhead appliances while retaining existing overlapping address spaces.

The concept of NSV originates in an MPLS VPN environment with multiple hosts in the same source VPN. The hosts require access to different servers in various destination VPNs. This is a difficult deployment to implement if a particular subinterface is VRF-attached. A subinterface is a way to partition configuration information for certain subsets of traffic that arrive or leave a physical interface.

NSV uses the IOS MPLS VPN VRF Select feature, which essentially eases the requirement of a VRF-attached subinterface.

The VRF Select feature uses policy-based routing (PBR) at the ingress interface of the VRF router to determine which VRF to forward traffic to. In most cases, the VRF router is a PE device. In a VRF-Lite implementation, the VRF router is a CE device. The VRF router determines the routing and forwarding of packets coming from the customer networks (or VPNs). The access control list (ACL) defined in the PBR route map matches the source IP address of the packet. If it finds a match, it sends the packet to the appropriate MPLS VPN (the VRF table).

For more information about PBR, see [“Policy-Based Routing Virtual In-Path Deployments”](#) on page 255.

The VRF table contains the virtual routing and forwarding information for the specified VPN. It forwards the selected VPN traffic to the correct MPLS label switched path (LSP), based upon the destination IP address of the packet.

NSV with VRF Select removes the association between the VRF and the subinterface. Decoupling the VRF and the subinterface allows you associate more than one MPLS VPN to the subinterface. The subinterface remains in the IPv4 dimension in VRF Select (as compared to the VPNv4 address space, in which it resides when it is VRF-attached). The subinterface is still IPv4-based, but it becomes aware of VRF Select by replacing the `ip vrf forwarding` Cisco command with `ip vrf receive` command.

The result is that the subinterface becomes *Not-So-VRF*. The subinterface still resides in the global IPv4 table, but it now uses PBR for the VRF switch. The PBR route map matches criteria based on traffic flows to be optimized.

IOS Requirements

Cisco recommends the following minimum IOS releases for a MLPS VPN VRF Selection using PBR deployment.

Cisco Hardware	Cisco IOS
Most Router Platforms	12.3(7)T or later
C76xx	12.2(33)SRB1, 12.2(33)SRB2, 12.2(33)SRC, 12.2(33)SRC1, 12.2(33)SRC2
ASR 1000 Series Router	XE 2.1.0, 2.1.1, 2.1.2, 2.2.1

Important: Regardless of how you configure a Steelhead appliance, if the Cisco IOS version on the router or switch is below the current Cisco minimum recommendations, it might be impossible to have a functioning NSV implementation, or the implementation might not have optimal performance.

Prerequisites for NSV

Before configuring NSV, review the following information:

- A detailed network diagram illustrating the logical connectivity between the data centers and branch offices
- A running configuration of the multiple VRF CE devices
- The exact IOS versions and hardware platforms in use

Example NSV Network Deployment

This example shows the following deployment scenario:

- One Steelhead appliance, configured as a logical in-path (data center)
- One Steelhead appliance, configured as a physical in-path (branch office)
- Both Steelhead appliances are running RiOS v5.0.3 or later
- The operating system is 12.3(15)T7
- Two units of 3640 series routers
- Two units of WinXP VM hosts
- IP Service Level Agreement (SLA)
- Static routes with tracking

This deployment is the basis for the configuration shown in [“Configuring NSV” on page 328](#).

Figure 14-3 shows a logical in-path NSV deployment in a VRF network environment.

Figure 14-3. Sample NSV Network Setup

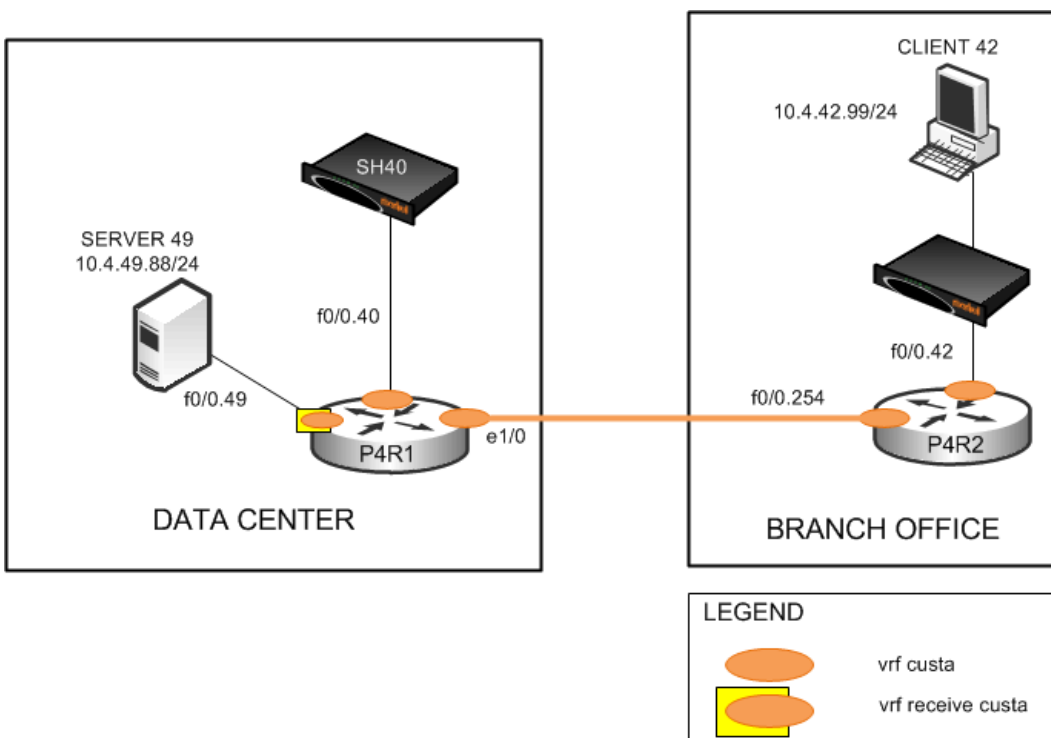


Figure 14-4 shows the NSV deployment shown in Figure 14-3 with intercepted and optimized flows.

Figure 14-4. NSV Deployment with Intercepted and Optimized Flows

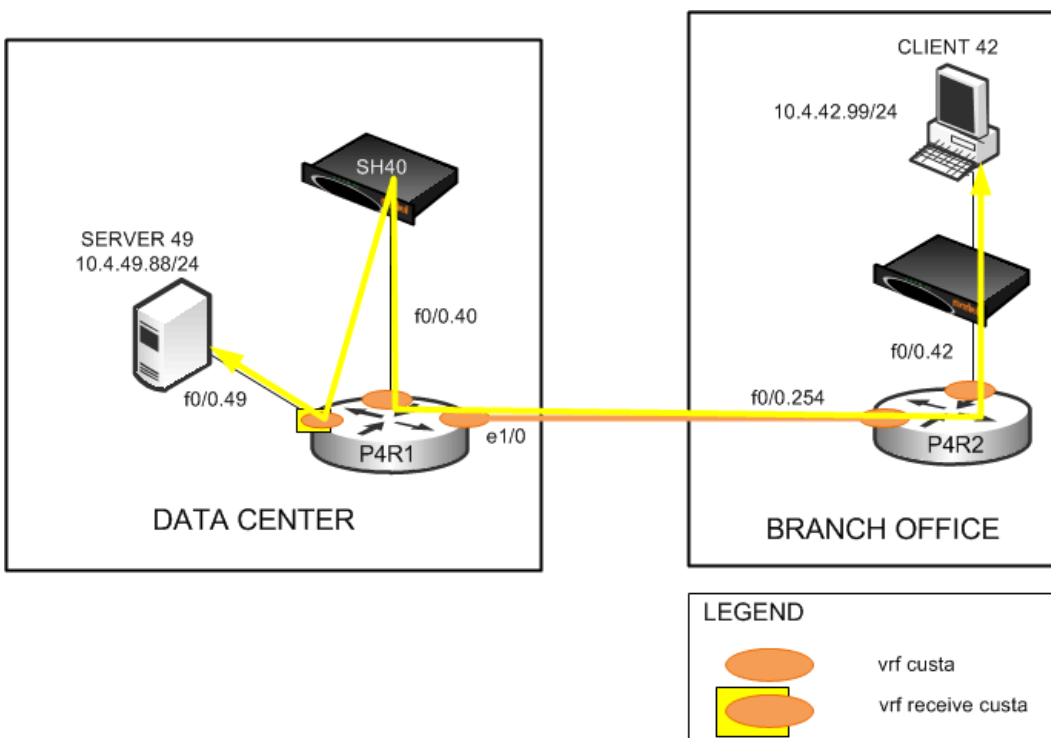
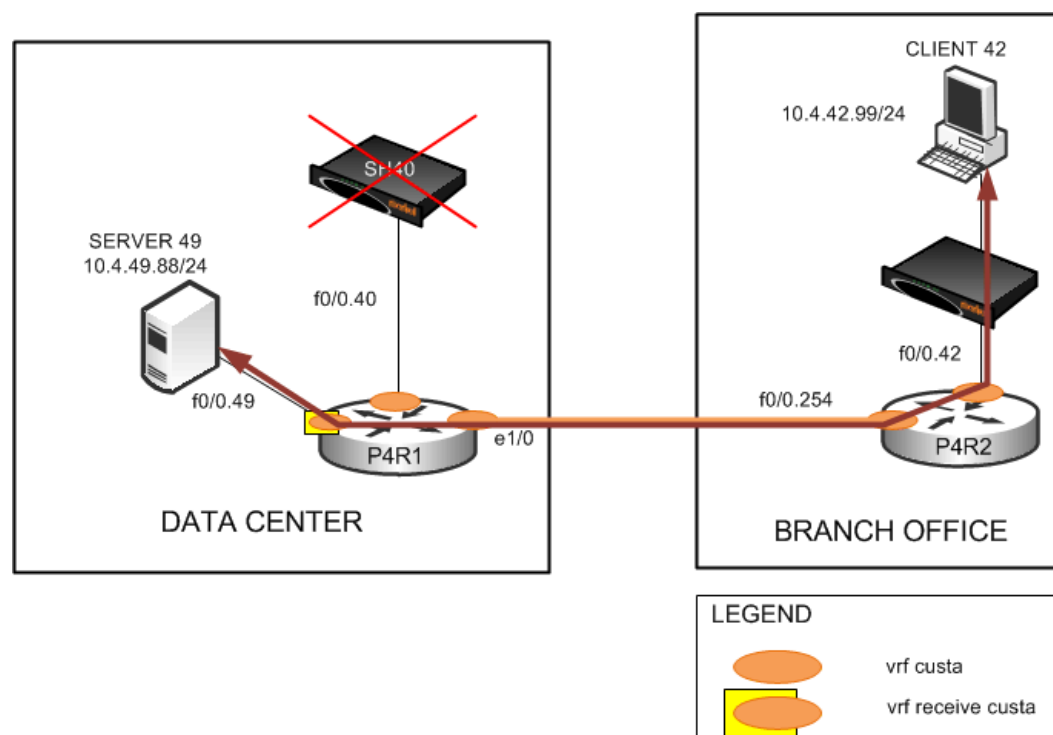


Figure 14-5 shows the NSV deployment with bypassed flows in the event that the data center Steelhead appliance fails.

Figure 14-5. NSV Deployment with Bypassed Flows



Configuring NSV

This section describes how to configure NSV. This section includes the following topics:

- [“Basic Steps for Configuring NSV” on page 329](#)
- [“Configuring the Data Center Router” on page 329](#)
- [“Configuring the PBR Route Map” on page 330](#)
- [“Decoupling VRF from the Subinterface to Implement NSV” on page 331](#)
- [“Configuring Static Routes” on page 331](#)
- [“Configuring the Branch Office Router ” on page 332](#)
- [“Configuring the Data Center Steelhead Appliance” on page 333](#)
- [“Configuring the Branch Office Steelhead Appliance ” on page 333](#)

The configuration in this section uses the parameters set up in [“Example NSV Network Deployment” on page 326](#).

Basic Steps for Configuring NSV

This section shows the overview of the basic steps to configure NSV with VRF Select. The following sections describe each step in detail.

To configure basic NSV with VRF select

1. Configure the data center PE or CE router, which includes defining:
 - the VRF tables.
 - the subinterfaces.
 - the PBR route map.
 - PBR.
 - static routes.
 - parameters for monitoring the Steelhead appliance availability.
2. Configure the branch office router.
3. Configure the data center Steelhead appliance.
4. Configure the branch office Steelhead appliance.

The following sections describe each of these steps in detail.

Configuring the Data Center Router

The data center PE or CE router determines the routing and forwarding of packets coming from the customer networks or VPNs. This device requires the most configuration.

The first step is to define the VRF tables for the Steelhead appliance. For example, you define two VRF tables for Steelhead appliance 40: `custa` for the customer and `wds_a` to use as a dummy VRF table. The dummy VRF table is not tied to any interface. It redirects traffic with a corresponding default route, which points to or exits at the subinterface to the Steelhead appliance.

Note: You cannot enter the `set ip next-hop` Cisco command on a PBR route map configured for VRF select.

The next step configures the subinterfaces and VRF routing protocol. In this example, you configure the following subinterfaces and define the OSPF VRF routing protocol:

- `f0/0.40` (the LAN-to-Steelhead appliance 40)
- `e1/0` (the WAN)

Note: This example uses OSPF as the routing protocol, but you can use other protocols, such as RIP, EIGRP, ISIS, and BGP, as well. OSPF uses a different routing process for each VRF. For the other protocols, a single process can manage all the VRFs.

To define the VRF tables and subinterfaces

1. Define the VRF tables for the Steelhead appliance. On the data center router (in this example, P4R1), enter the following commands:

```
hostname p4R1
!
ip cef
!
ip vrf custa
    rd 4:1
    !
ip vrf wds_a
    rd 4:9
    !
```

2. Configure the VRF subinterfaces and corresponding VRF routing protocol. On the data center router, at the system prompt, enter the following commands:

```
interface FastEthernet0/0.40
    encapsulation dot1Q 40
    ip vrf forwarding custa
    ip address 10.4.40.1 255.255.255.0
    !
interface Ethernet1/0
    ip vrf forwarding custa
    ip address 10.254.4.1 255.255.255.0
    half-duplex
    !
    router ospf 4 vrf custa
    redistribute static subnets
    network 10.4.40.0 0.0.0.255 area 0
    network 10.254.4.0 0.0.0.255 area 0
```

This example configures the LAN subinterface f0/0.40, which interconnects Steelhead appliance 40 to use VRF custa. Later, you point the dummy VRF wds_a to a default route (in this example, f0/0.40). This enables a PBR route map at f0/0.49 to redirect incoming traffic from Server 49 to Client 42 to Steelhead appliance 40 for optimization.

In this example, because Client 42 is in the VPN custa (VRF custa), the traffic must return to the VRF custa routing path after optimization. For this redirection to work, the Steelhead appliance 40 must reside in VRF custa and *not* VRF wds_a.

Configuring the PBR Route Map

VRF Select requires a control mechanism such as PBR to select which particular VRF table a data packet goes to. The next step is to configure a PBR route map, which provides a matching criteria for incoming traffic and sets the VRF table.

To configure the PBR route map

- On the data center router, enter the following commands:

```
route-map wds_a permit 10
    match ip address 104
    set vrf wds_a
    !
route-map wds_a permit 20
    set vrf custa
    !
access-list 104 permit tcp host 10.4.49.88 host 10.4.42.99
```

The route map `wds_a` matches incoming traffic from Server 49 to Client 42. When it finds a match, it sets the VRF to `wds_a`, which, in turn, points to default route `f0/0.40`, where Steelhead appliance 40 resides. Binding `f0/0.40` with VRF `custa` ensures that the returning optimized traffic eventually reaches Client 42. The route map also sets any incoming traffic to VRF `custa`—except Server 49 to Client 42.

Important: Ensure that the PBR route map contains a default `set vrf` in the PBR route map to always match a packet that does not match any of the previous criteria.

Important: Because BGP control packets are required to remain categorized as global-IPv4, use an ACL to ensure that these packets do not get forwarded to a VRF table.

Decoupling VRF from the Subinterface to Implement NSV

The following step decouples the association between VRF and a subinterface. It implements NSV by replacing the `ip vrf forwarding` Cisco command with `ip vrf receive` command.

The result is that the subinterface becomes *Not-So-VRF*. The subinterface still resides in the global IPv4 table, but it now uses PBR for the VRF switch. The PBR route map matches criteria based on traffic flows to be optimized.

Important: You must have already defined the PBR route map as described in [“Configuring the PBR Route Map” on page 330](#) before completing the next step.

To implement VRF Select and PBR

- On the data center router, enter the following commands:

```
interface FastEthernet0/0.49
  encapsulation dot1Q 49
  ip vrf receive custa
  ip address 10.4.49.1 255.255.255.0
  ip policy route-map wds_a
```

The absence of the `ip vrf forwarding` command in this example configuration implies that `f0/0.49` is not associated with any particular VRF and remains in the IPv4 global address space. This makes it possible for the Steelhead appliances to communicate with the subinterface.

Configuring Static Routes

Static routes play a crucial role in an NSV deployment, because you use them to fine-tune the routing. The primary, default static route points to the in-path interface to redirect incoming traffic for optimization. (In the following example, traffic is redirected to 10.4.40.101 of Steelhead appliance 40).

The command keyword `track 1` determines whether the in-path IP address of the Steelhead appliance is reachable. The primary, default static route is used only when the in-path IP address for the Steelhead appliance is reachable. If it becomes unreachable, the primary route is removed from the routing table. The second, floating route serves as a backup to avoid blackholing traffic and ensure flow continuity.

In this example, when the primary route is removed from the routing table because the Steelhead appliance is unreachable, the second route becomes effective at an administrative weight of 250, points to the WAN interface `e1/0`, and avoids blackholing traffic to ensure flow continuity.

Also, in this example, because f0/0.49 (where Server 49 is connected) is still in the IPv4 global address space, you must make it visible in VRF custa. To do this, you assign a third static route, associating it with VRF custa. The third static route points to Server 49 (10.4.49.88) in VRF custa and redistributes it into OSPF.

To define static routes

- On the data center router, enter the following commands:

```
ip route vrf wds_a 0.0.0.0 0.0.0.0 FastEthernet0/0.40 10.4.40.101 track 1
ip route vrf wds_a 0.0.0.0 0.0.0.0 Ethernet1/0 10.254.4.2 250
ip route vrf custa 10.4.49.88 255.255.255.255 FastEthernet0/0.49 10.4.49.88
!
```

To monitor Steelhead appliance availability

- On the P4R1, at the system prompt, enter the following commands:

```
ip sla monitor 1
  type echo protocol ipIcmpEcho 10.4.40.101
  vrf custa
  frequency 5
!
ip sla monitor schedule 1 life forever start-time now
!
track 1 rtr 1 reachability
```

IP SLA uses the ICMP echo protocol to monitor the availability status of the Steelhead appliance in-path IP address every 5 seconds (in this example, IP address 10.4.40.101 for Steelhead appliance 40:custa). This is tied to the primary default route through the tracking mechanism. The tracking mechanism prevents routing to an unavailable IP destination when the in-path IP address for the Steelhead appliance is down (in this example, Steelhead 40:custa).

Configuring the Branch Office Router

A typical branch office router is a PE VRF or CE VRF Lite device. Its configuration is minimal and standard. In most environments you probably do not need to configure this device.

To configure the P4R2

- On the P4R2, enter the following commands:

```
hostname P4R2
ip cef
ip vrf custa
  rd 4:1
interface FastEthernet0/0.42
  encapsulation dot1Q 42
  ip vrf forwarding custa
  ip address 10.4.42.1 255.255.255.0
interface FastEthernet0/0.254
  encapsulation dot1Q 254
  ip vrf forwarding custa
  ip address 10.254.4.2 255.255.255.0
router ospf 4 vrf custa
  network 10.4.42.0 0.0.0.255 area 0
  network 10.254.4.0 0.0.0.255 area 0
```

Configuring the Data Center Steelhead Appliance

The data center Steelhead appliance (in this example, VRF *custa*) is another vital component of an NSV deployment. Its configuration is very simple; you simply enable the logical in-path interface.

To configure the data center Steelhead appliance

- On the data center Steelhead appliance, connect to the CLI and enter the following commands:

```
hostname "SH40"  
interface inpath0_0 ip address 10.4.40.101 /24  
ip in-path-gateway inpath0_0 "10.4.40.1"  
in-path enable  
in-path oop enable  
write memory  
restart
```

Note: You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

Configuring the Branch Office Steelhead Appliance

The Steelhead appliance deployed at the branch office needs slightly more configuration than the data center Steelhead appliance. Because you are only implementing VRF Select for redirecting the data center LAN-side traffic, you must define fixed-target rules for the WAN-side traffic.

The following example uses Steelhead 42:*custa*.

To configure the branch office Steelhead appliance:

- On the branch office Steelhead appliance, connect to the CLI and enter the following commands:

```
hostname "SH42"  
interface inpath0_0 ip address 10.4.42.101 /24  
ip default-gateway "10.4.42.1"  
in-path enable  
in-path rule fixed-target target-addr 10.4.40.101 target-port 7800 dstaddr  
10.4.49.88/32 dstport "all" srcaddr 10.4.42.99/32 rulenum 4
```

Note: You must save your changes or they are lost upon reboot. Restart the optimization service for the changes to take effect.

You can also use auto-discovery to eliminate configuring fixed-target rules if you disassociate the WAN interface (in this example, P4R1 e1/0) from the VRF (in this example, *custa*) the same way you disassociated the LAN interface using VRF select, as described in [“Decoupling VRF from the Subinterface to Implement NSV” on page 331](#).

The branch office Steelhead appliance could also be a Mobile Client. In this deployment, you could use the Mobile Controller to facilitate configuring the fixed-target rules.

For details on the Mobile Controller, see the *Steelhead Mobile Controller User's Guide*.

CHAPTER 15 Out-of-Path Deployments

This chapter describes out-of-path deployments and summarizes the basic steps for configuring them. This chapter includes the following sections:

- [“Overview of Out-of-Path Deployment” on page 335](#)
- [“Limitations of Out-of-Path Deployments” on page 336](#)
- [“Configuring Out-of-Path Deployments” on page 337](#)

For details on the factors you must consider before you design and deploy the Steelhead appliance in a network environment, see [“Choosing the Right Steelhead Appliance” on page 16](#).

Note: Riverbed refers to WCCP and PBR deployments as virtual in-path deployments. This chapter discusses out-of-path deployments, which do not include WCCP or PBR deployments.

This chapter requires that you be familiar with the installation and configuration process for the Steelhead appliance. For details, see the *Steelhead Appliance Installation and Configuration Guide*.

Overview of Out-of-Path Deployment

In an out-of-path deployment, only a Steelhead appliance primary interface is required to connect to the network. The Steelhead appliance can be connected anywhere in the LAN. An out-of-path Steelhead appliance deployment does not have a redirecting device. You configure fixed-target in-path rules for the client-side Steelhead appliance. The fixed-target in-path rules point to the primary IP address of the out-of-path Steelhead appliance. The out-of-path Steelhead appliance uses its primary IP address when communicating to the server. The remote Steelhead appliance must be deployed either in a physical or virtual in-path mode.

[Figure 15-1](#) shows an out-of-path deployment.

An out-of-path deployment is generally located on the server-side and is often described as a *server-side out-of-path deployment*.

You can achieve redundancy by deploying two Steelhead appliances out-of-path at one location, and by using both of their primary IP addresses in the remote Steelhead appliance fixed-target rule. The fixed-target rule allows the specification of a primary and a backup Steelhead appliance. If the primary Steelhead appliance becomes unreachable, the remote Steelhead appliances use the backup Steelhead appliance until the primary comes back online. If both out-of-path Steelhead appliances in a specific fixed-target rule are unavailable, the remote Steelhead appliance passes through this traffic unoptimized. The remote Steelhead appliance does not look for another matching in-path rule in the list.

You can use RiOS data store synchronization between the out-of-path Steelhead appliances for additional benefits in case of a failure. For details, see [“RiOS Data Store Synchronization” on page 15](#).

You can also implement load balancing with out-of-path deployments by using multiple out-of-path Steelhead appliances, and configuring different remote Steelhead appliances to use different target out-of-path Steelhead appliances.

You can target an out-of-path Steelhead appliance for a fixed-target rule. This can be done simultaneously for physical in-path and virtual in-path deployments. This is referred to as a *hybrid deployment*.

For details on fixed-target in-path rules, see [“Fixed-Target In-Path Rules” on page 33](#).

Limitations of Out-of-Path Deployments

Although the ease of deploying an out-of-path Steelhead appliance might seem appealing, there are serious disadvantages to this method:

- Connections initiated from the site with the out-of-path Steelhead appliance cannot be optimized.
- Servers at the site detect the optimized traffic coming not from a client IP address, but from the out-of-path Steelhead appliance primary IP address.

In certain network environments, a change in the source IP address can be problematic. For some commonly used protocols, Steelhead appliances automatically make protocol-specific adjustments to account for the IP address change. For example, with CIFS, MAPI, and FTP, there are various places where the IP address of the connecting client can be used within the protocol itself. Because the Steelhead appliance uses application-aware optimization for these protocols, it is able to make the appropriate changes within optimized connections and ensure correct functioning when used in out-of-path deployments. However, there are protocols, such as NFS, that cannot function appropriately when optimizing in an out-of-path configuration.

Important: If you use out-of-path deployments, ensure correct operation by carefully selecting which applications you optimize. Even with protocols in which RiOS specifically adjusts for the change in source IP address on the LAN, there might be authentication, IDS, or IPS systems that generate alarms when this change occurs.

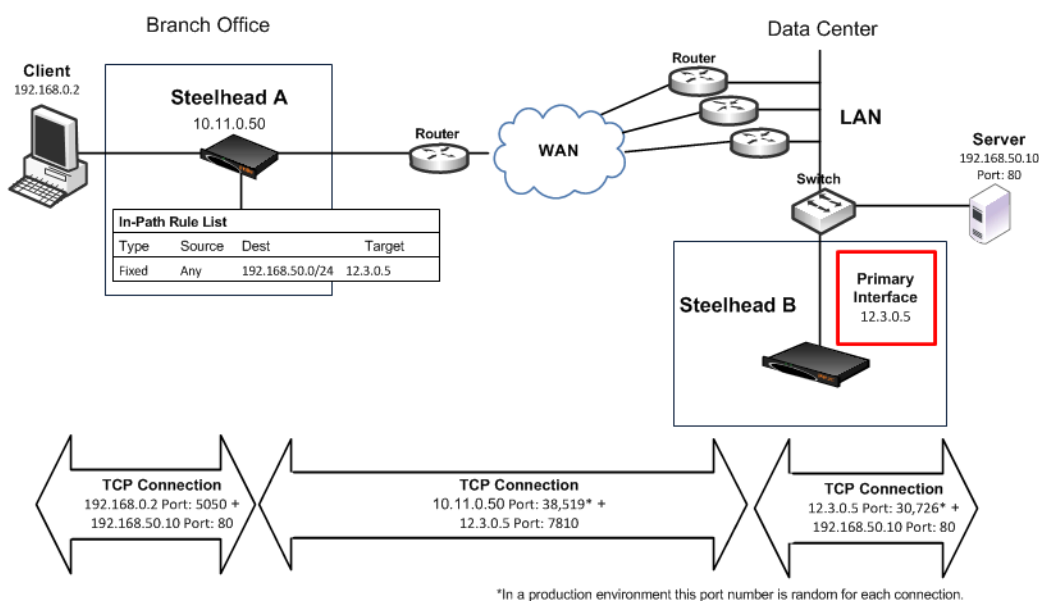
Because of the disadvantages specific to out-of-path deployments, and the requirement of using fixed-target rules, out-of-path deployment is not as widely used as physical or virtual in-path deployments. Out-of-path is primarily used as a way to rapidly deploy a Steelhead appliance in a site with very complex or numerous connections to the WAN.

Configuring Out-of-Path Deployments

Figure 15-1 shows a scenario in which fixed-target in-path rules are configured for an out-of-path Steelhead appliance primary interface.

Note: This section provides the basic steps for out-of-path network deployments. It does not provide detailed procedures. Use this section as a general guide.

Figure 15-1. A Fixed-Target In-Path Rule to an Out-of-Path Steelhead Appliance Primary IP Address



In this example, you configure:

- Steelhead A with a fixed-target in-path rule specifying that traffic destined to a particular Web server at the data center is optimized by the out-of-path Steelhead B.
- The TCP connection between the out-of-path Steelhead appliance, Steelhead B, and the server uses the Steelhead appliance primary IP address as the source, instead of the client IP address.

To configure a basic out-of-path deployment

1. On Steelhead A, connect to the CLI and enter the following commands:

```
enable
configure terminal
in-path rule fixed-target target-addr 12.3.0.5 dstaddr 192.168.50.0/24 dstport 80 rulenum end
```

2. On Steelhead B, connect to the CLI and enter the following commands:

```
enable
configure terminal
out-of-path enable
```


CHAPTER 16 Data Protection Deployments

This chapter describes the configuration and deployment of Steelhead appliances for data protection solutions. By leveraging Steelhead appliances, you can achieve higher levels of data protection, streamlined IT operations, and reduce WAN bandwidth.

This chapter includes the following sections:

- [“Overview of Data Protection” on page 339](#)
- [“Planning for a Data Protection Deployment” on page 340](#)
- [“Configuring Steelhead Appliances for Data Protection” on page 345](#)
- [“Common Data Protection Deployments” on page 349](#)
- [“Designing for Scalability and High Availability” on page 351](#)
- [“SnapMirror Optimization” on page 353](#)
- [“Troubleshooting and Fine-Tuning” on page 354](#)
- [“Third-Party Interoperability” on page 355](#)

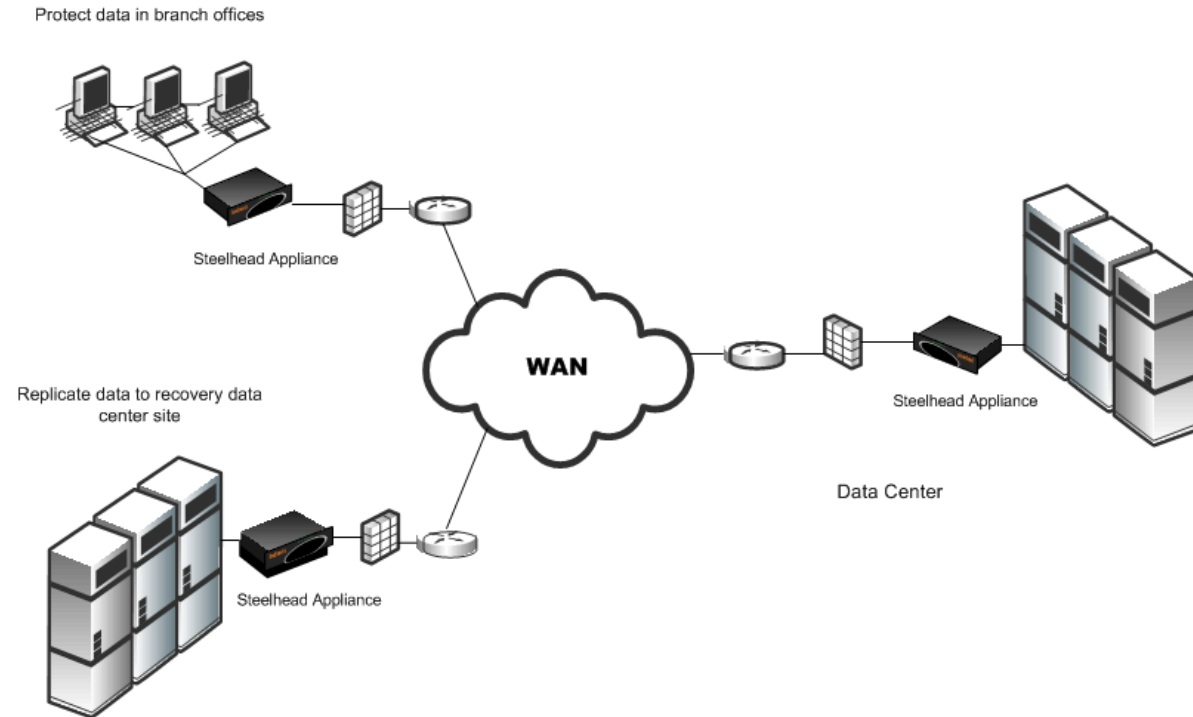
Overview of Data Protection

To secure and recover important files and data, more data center-to-data center environments (or branch office-to-data center environments) are using WAN-based backup and data replication (DR). WAN optimization is now a critical part of data protection environments because it can substantially reduce the time it takes to replicate data, perform backups, and recover data. Backup and replication over the WAN ensures that you can protect data safely at a distance from the primary site, but it can also introduce new performance challenges. To meet these performance challenges, Riverbed provides hardware and software capabilities that help data protection environments in the following ways:

- **Reduce WAN Bandwidth** - By reducing WAN bandwidth, Steelhead appliances can lower the total cost of current data protection procedures and, in some cases, make WAN-based backup or replication possible where it was not before.

- **Accelerate Data Transfer** - By accelerating data transfer, Steelhead appliances meet or improve time targets for protecting data.

Figure 16-1. A Data Protection Deployment Using WAN-Based Replication



Planning for a Data Protection Deployment

This section describes methods for planning a successful data protection deployment. You must consider several variables, each of which can have a significant impact on the model, number, and configuration of Steelhead appliances required to deliver the required result. This section includes the following topics:

- [“LAN-side Throughput and Data Reduction Requirements” on page 340](#)
- [“Predeployment Questionnaire” on page 342](#)

Riverbed strongly recommends that you read both of these sections and complete the questionnaire. Riverbed also recommends that you consult with Riverbed Professional Services or an authorized Riverbed Delivery Partner when planning for a data protection deployment.

For details on the other factors to consider before you design and deploy the Steelhead appliance in a network environment, see [“Choosing the Right Steelhead Appliance” on page 16](#).

LAN-side Throughput and Data Reduction Requirements

This section describes requirements and configurations from LAN-side throughput and data reductions. This section includes the following topics:

- [“Configuring a Nightly Full Database Backup” on page 341](#)
- [“Configuring a Daily File Server Replication” on page 341](#)

- [“Configuring a Very Large Nightly Incremental Backup” on page 342](#)

The basis for correctly qualifying, sizing, and configuring Steelhead appliances for use in a data protection environment depends on that the deployed Steelhead appliances can:

- receive and process data on the LAN at the required rate (LAN-side throughput), and
- reduce the data by a certain X-Factor, to
- transfer data given certain WAN-side bandwidth constraints.

These constraints are defined by the following formula:

$$\text{LAN-side Throughput} / \text{X-Factor} \leq \text{WAN-side Bandwidth}$$

You derive the LAN-side throughput requirements from an understanding of the maximum amount of data that must be transferred during a given time period. Often, the time allotted to transfer data is defined as a target Recovery Point Objective (RPO) for your organization.

The RPO describes the acceptable amount of data loss measured in time. You must recover data at this time. This is generally a definition of what an organization determines is an acceptable data loss following a disaster; it is measured in seconds, minutes, hours, days, or weeks. For example, an RPO of two hours means that you can always recover the state of data two hours in the past.

Note: The following link provides an Excel throughput calculator that you can use to calculate bandwidth requirements expressed in other forms of time objectives: <https://splash.riverbed.com/message/8478#8478>.

The X-Factor describes the level of data reduction necessary to fit the LAN data into the WAN link. For example, if LAN-side throughput required to meet RPO is 310 Mbps and WAN-side bandwidth available is 155 Mbps, then X-Factor is 2x. X-Factor is highly dependent on the nature of the data, but in practice it generally ranges from 2x (for LZ-only compression) to 4-8x (for default SDR mode).

Configuring a Nightly Full Database Backup

Objective:

“I want to copy 1.8 TB of nightly database dumps over my OC-3 within a 10-hour window.”

Formula:

$$1.8 \text{ TB} / 10 \text{ hours} = 400 \text{ Mbps}$$

Solution:

An OC-3 link has a capacity of 155 Mbps. To deliver 400 Mbps, the Steelhead appliance must reduce the total bandwidth over the WAN by $400/155 = 2.58x$.

Configuring a Daily File Server Replication

Objective:

“After consolidating the NetApp file servers from branch offices, I expect daily SnapMirror updates from my data center to go from 400 GB to 4 TB per day. I have a designated DS-3 that is nearly maxed out. Can the Steelhead appliance help me replicate all 4 TB each day using my DS-3?”

Formula:

$$4 \text{ TB} / 1 \text{ day} = 370 \text{ Mbps}$$

Solution:

A DS-3 link has a capacity of 45 Mbps. To deliver 370 Mbps, the Steelhead appliance must reduce the total bandwidth over the WAN by $370/45 = 8.2x$. This is within the range of data reduction that the Steelhead appliance can achieve using default SDR, depending on the amount of redundancy present in the data streams.

Configuring a Very Large Nightly Incremental Backup

Objective:

“The incremental Tivoli Storage Manager (TSM) backup at a remote site is typically 600 GB and the backup window each night is 8 hours. Can I perform these backups over the WAN using a T1 link?”

Formula:

$$600 \text{ GB} / 8 \text{ hours} = 166 \text{ Mbps}$$

Solution:

A T1 link has a capacity of 1.5 Mbps. To deliver 166 Mbps, the Steelhead appliances must reduce the total bandwidth over the WAN by $166/1.5 = 110x$. This is a very high level of reduction that is typically out of range for data protection deployments.

To support backups over the WAN, you must upgrade the WAN link. A T3 link, for example, has a capacity of 45 Mbps. Using a T3 link, the Steelhead appliances needs to achieve a data reduction of $166/45 = 3.7x$, which is attainable for many deployments.

Predeployment Questionnaire

To organize and take a survey of the WAN-side, LAN-side, and X-Factor considerations, use the predeployment questionnaire in the following table. Discuss your completed survey with Riverbed Professional Services or an authorized delivery partner, to determine the best model, number, and initial configuration of the Steelhead appliances to deploy.

For a Microsoft Word version of the Data Protection questionnaire go to <http://splash.riverbed.com/message/3194>.

Question	Why This Is Important
WAN-side Considerations	
Is this a two-site or a multisite (fan-in, fan-out) data protection opportunity?	In a two-site deployment, the same Steelhead appliance models are often selected for each site. In a multisite (fan-in, fan-out) deployment, the Steelhead appliance at the central site is sized to handle the data transfers to and from the edge sites.
What is the WAN link size?	Knowing the WAN link size is essential in determining: <ul style="list-style-type: none"> • which models are feasible for deployment because the Steelhead appliances specifications are partially based on the WAN rating. • the level of data reduction the Steelhead appliances must deliver to meet the ultimate data protection objective.
What is the network latency between sites?	Knowing the latency in the environment is essential for providing accurate performance estimates. Network latency and WAN link size are used together to calculate buffer sizes on the Steelhead appliance to provide optimal link utilization. Although Steelhead appliances are generally able to overcome the effects of latency for network protocols used in data protection solutions, some are still latency sensitive.

Question	Why This Is Important
Is there a dedicated link for disaster recovery?	Environments with a dedicated link are typically easier to configure. Environments with shared links must employ features such as QoS to ensure that data protection traffic receives an adequate amount of bandwidth necessary to meet the ultimate objective.
LAN-side Considerations	
Which backup or replication products are you using?	<p>Certain backup or replications products require special configuration. Knowing what is currently in use is essential for providing configuration recommendations and performance estimates. Riverbed has experience with different data protection products and business relationships with many different replication vendors. Many have similar configuration options and network utilization behaviors.</p> <p>Some examples of backup and replication products:</p> <ul style="list-style-type: none"> • EMC - SRDF/A RecoverPoint • NetApp - SnapMirror • Symantec - NetBackup • Vision Solutions - Double-Take • CA - ARCserve • HP - Continuous Access EVA • HDS - TrueCopy • IBM - PPRC
Are you using synchronous or asynchronous replication?	<p>Asynchronous replication is typically a very good fit. By comparison, synchronous replication has very stringent latency requirements and is rarely a good fit for WAN optimization.</p> <p>Many types of data protection traffic are not typically considered <i>replication</i> of either type, such as backup jobs.</p>
What is your backup methodology?	<p>Knowing the backup type and schedule provides insight into the frequency of heavy data transfers and the level of repetition within these transfers.</p> <p>Some examples of backup methodologies are:</p> <ul style="list-style-type: none"> • A single full backup and an incremental backup for life (synthetic full). • A daily full backup. • A weekly full backup and a daily incremental backup.
Are your data streams single or multistream? What is the total number of replication streams?	<p>Knowing the number of TCP streams is essential in providing a configuration recommendation and performance estimate. Because Steelhead appliances proxy TCP/IP, the number of TCP streams created by the data protection solution can impact the Steelhead appliance resource utilization.</p> <ul style="list-style-type: none"> • RiOS v5.0 and earlier have a constraint that each TCP session (<i>stream</i>) is serviced by a single CPU core, so splitting the load across many streams is essential to fully use the resources in larger, multicore Steelhead appliances. • RiOS v5.5 or later has multicore features that allow multiple CPU cores to process a single stream. <p>When considering the number of streams, of primary importance is the number of heavyweight data streams that carry significant amounts of traffic. In addition, consider that any smaller control streams that carry a small amount of traffic (such as these present in many backup systems and some FCIP systems).</p> <p>Finally, depending on the data protection technology in use, there might be options to increase the number of streams in use. As a first step, determine how many streams are observed in the current environment. Determine whether there is a willingness to increase the number of data streams if a method to do so is suggested.</p>

Question	Why This Is Important
<p>Is there a FCIP/iFCP gateway?</p> <p>If yes, what is the make, model, and firmware version?</p>	<p>Some FCIP/iFCP gateways (or particular firmware versions of some gateways) do not adhere fully to the TCP/IP or FCIP standards. Depending on what is in use they might require firmware upgrades, special configuration, or cannot be optimized at this time.</p> <p>Gateways are mainly seen in fibre channel SAN replication environments such as SRDF/A, MirrorView, and TrueCopy.</p> <p>Typical firmware versions: Cisco MDS, FCIP v4.1(3) Brocade 7500 FOS v6.3.1, QLogic isr6142 v2.4.3.2.</p>
<p>Is compression enabled on the gateway or the replication product?</p> <p>If yes, what is the current compression ratio?</p>	<p>Most data protection environments using FCIP or iFCP gateways use their built-in compression method, as this is a best practice of the product vendors and the SAN vendors who configure them. However, the best practice for WAN optimization of these technologies is to disable any compression currently in use and employ the Steelhead appliance optimization instead.</p> <p>The first-pass LZ compression in the Steelhead appliance typically matches the compression already in use and then RiOS SDR allows for an overall level of data reduction that improves the previous compression ratio.</p> <p>Knowing the current compression ratio achieved using the built-in compression method is important in determining whether the Steelhead appliances can improve upon it.</p>
<p>Are Steelhead appliances already deployed?</p> <p>If yes, what is their make and the RiOS version?</p>	<p>If the environment already has Steelhead appliances deployed and data protection is a new requirement, knowing the current appliance models in use can determine if adequate system resources are available to meet the objectives without adding additional hardware.</p> <p>Knowing the current RiOS version is essential in determining what features and tuning opportunities are available in the RiOS release to provide the optimal configuration for data protection. If the environment does not already use Steelhead appliances, Riverbed can recommend the ideal RiOS version based on the environment and data protection objective.</p>
X-Factor Considerations	
<p>How much new incremental data is added daily or hourly?</p>	<p>The rate of change information is extremely useful alongside the dataset size information to provide accurate performance estimates. If a dataset is too large for a single RiOS data store to find the data patterns for the entire dataset without wrapping continuously, Riverbed can plan system resources based on servicing the amount of data that changes hourly or daily.</p>
<p>What is the total size of the dataset?</p>	<p>For some data protection solutions such as backup, knowing the dataset size is extremely important for RiOS data store sizing. Ideally you want to select Steelhead appliances that can find the data patterns for the entire dataset without continuously wrapping the RiOS data store.</p> <p>For SAN-based solutions this information can be more difficult to gather, but even rough estimates can help. For example, you can estimate the size of the Logical Unit Number (LUNs) that are subject to replication or the size of the databases stored on an array.</p>
<p>What is the dataset type? For example, Exchange, VMware, SQL, or file system.</p>	<p>Different types of data exhibit different characteristics when they appear on the network as backup or replication traffic. For example, file system data or VMware images often appear as large, sequential bulk transfers, and lend themselves well to disk-based data reduction.</p> <p>On the other hand, real-time replication of SQL database updates can often present a workload that requires heavy amounts of disk seeks. These types of workloads can lend themselves better to a memory-based approach to data reduction.</p>
<p>Is the data pre-compressed?</p>	<p>You must determine if precompressed data is present for accurate performance estimates. Data stored at the point of origin in a precompressed format (such as JPEG images, video, or any type of data that has been compressed separately with utility tools such as WinZip), might see limited data reduction from Steelhead appliances.</p>

Question	Why This Is Important
Is the data encrypted?	Data stored at the point of origin in a pre-encrypted format (such as DPM-protected documents or encrypted database fields and records) might see limited data reduction from the Steelhead appliance.
How repeatable is the data?	You must determine if repeatable data is present for accurate performance estimates. Data that contains internal repetition (such as frequent, small updates to large document templates) typically provide very high levels of data reduction.
What LAN-side throughput is needed to meet the data protection goal?	It is the speed of data going in and out of the systems on the LAN that establishes whether the data protection objectives can be met. The LAN-side throughput can be calculated by dividing the total amount of changed data by the time window for the replication or backup job. The WAN-side throughput and level of data reduction represent the level of optimization.

Configuring Steelhead Appliances for Data Protection

After you deploy the Steelhead appliances and perform the initial configuration, you can use the features described in this section to deliver an optimal data protection deployment. This section includes the following data protection features:

- [“Adaptive Data Streamlining Feature Settings” on page 346](#)
- [“CPU Settings” on page 346](#)
- [“Best Practices for Data Streamlining and Compression” on page 347](#)
- [“MX-TCP Settings” on page 348](#)
- [“The Steelhead Appliance WAN Buffer Settings” on page 348](#)
- [“The Router WAN Buffer Settings” on page 348](#)

You can configure the Steelhead appliance features relevant to data protection in the Management Console in the **Configure > Optimization > Performance** page.

Figure 16-2. Performance Page Data Protection Features

The screenshot displays the 'Performance' configuration page in the Steelhead Management Console. It is divided into two main sections: 'Adaptive Data Streamlining Modes' and 'CPU Settings'.

Adaptive Data Streamlining Modes: This section has two tabs: 'Maximize Data Reduction' (selected) and 'Maximize LAN Throughput'. Under the selected tab, there are four radio button options: 'Default' (selected), 'SDR-Adaptive', 'Legacy', and 'Advanced'.

CPU Settings: This section includes a 'Compression Level' dropdown menu set to 'Default'. Below it are two checkboxes: 'Adaptive Compression' (checked) and 'Multi-Core Balancing' (unchecked).

An 'Apply' button is located at the bottom left of the configuration area.

Adaptive Data Streamlining Feature Settings

Adaptive data streamlining provides you with the ability to fine tune the data streamlining capabilities, and enables you to obtain the right balance between optimal bandwidth reduction and optimal throughput.

The following table describes the adaptive data streamlining settings.

Adaptive Data Streamlining Setting	Benefit	Description
Default	Excellent bandwidth reduction	By default, Steelhead appliances use their disk-based RiOS data store to find data patterns that traverse the network. Previously seen data patterns do not traverse the network in their fully-expanded form. Instead, a Steelhead appliance sends a unique identifier for the data to its peer Steelhead appliance, which sends the fully-expanded data. In this manner, data is streamlined over the WAN because unique content only traverses the link once. The RiOS disk-based data store is able to maintain a large dictionary of segments and identifiers.
RiOS SDR-Adaptive	Good bandwidth reduction and LAN-side throughput	<p>Dynamically blends different data streaming modes to enable sustained throughput during periods of high disk-intensive workloads.</p> <p>Legacy: Monitors disk I/O response times, and based on statistical trends, employs a blend of disk-based de-duplication and compression-based data reduction techniques.</p> <p>Important: Use caution with this setting, particularly when optimizing CIFS or NFS with pre-population. For more information, contact Riverbed Support.</p> <p>Advanced: Monitors disk I/O response times and WAN utilization, and based on statistical trends, employs a blend of disk-based de-duplication, memory-based de-duplication and compress-based data reduction techniques.</p>
RiOS SDR-M	Excellent LAN-side throughput	<p>Performs data reduction entirely in memory, which prevents the Steelhead appliance from reading and writing to and from the disk. Enabling this option can yield high LAN-side throughput because it eliminates all disk latency. RiOS SDR-M is typically the preferred configuration mode for SAN replication environments.</p> <p>RiOS SDR-M is most efficient between two identical high-end Steelhead appliance models; for example, 6050 - 6050. When SDR-M is configured between two different Steelhead appliance models, the smaller model limits the performance.</p> <p>Important: When you use RiOS SDR-M, RiOS data store synchronization is not possible because none of the data is written to the disk-based data store. For details on data store synchronization, see “RiOS Data Store Synchronization” on page 15.</p>

CPU Settings

CPU settings provide you with the ability to balance throughput with the amount of data reduction and balance the connection load. The CPU settings are useful with high-traffic loads to scale back compression, increase throughput, and maximize Long Fat Network (LFN) utilization. This section includes the following topics:

- [“Compression Level” on page 347](#)
- [“Adaptive Compression” on page 347](#)
- [“Multicore Balancing” on page 347](#)

Compression Level

The compression level specifies the relative trade-off of LZ data compression for LAN throughput speed. Compression levels 1-9 can be specified for fine-tuning. Generally, a lower number provides faster throughput and slightly less data reduction. Setting the optimal compression level provides greater throughput although maintaining acceptable data reduction.

Riverbed recommends setting the compression to level 1 in high-throughput environments such as data-center-to-data-center replication.

Adaptive Compression

The adaptive compression feature detects the LZ data compression performance for a connection dynamically and turns it off (that is, sets the compression level to 0) momentarily if it is not achieving optimal results. Enabling this feature can improve end-to-end throughput in cases where the data streams are not further compressible.

Multicore Balancing

Multicore balancing distributes the load across all CPUs, therefore maximizing throughput. Multicore balancing improves performance in cases where there are fewer connections than the total number of CPU cores on the Steelhead appliance. Without multicore balancing, the processing of a given connection is bound to a single core for the life of the connection. With multicore balancing, even a single connection leverages all CPU cores in the system.

If you enable multicore balancing in cases where there are a large number of connections, there are no adverse effects. For this reason, you can enable multicore balancing in most (if not all) scenarios.

Best Practices for Data Streamlining and Compression

Riverbed recommends the following best practices for data protection scenarios:

- When replicating database log files, the LZ-only compression level typically provides optimal results because database log files contain few repetitive data sequences that can be deduplicated using RiOS SDR.
- Replication of databases like Exchange and Oracle typically works with RiOS SDR, but high-throughput environments can require additional configuration.
- For SAN replication environments (especially with high bandwidth), start with an RiOS SDR-M setting and deploy the same model Steelhead appliance on each side.
For details on SAN replication deployments, see [“Storage Area Network Replication” on page 350](#).
- Always set the compression level to 1 in high-throughput data center-to-data center replication scenarios.
- After the initial configuration, you can monitor disk performance by reviewing the RiOS Data Store Disk Load report accessible from the Management Console. The RiOS Data Store Disk Load report shows 100% for a sustained time or multiple times a day that coincide with periods of lower performance.

If you see symptoms of disk performance issues, switch to RiOS SDR-Adaptive mode to alleviate disk pressure.

For more details on best practice guidelines and configuration settings, see [“Common Data Protection Deployments” on page 349](#).

MX-TCP Settings

Maximum TCP (MX-TCP) enables data flows to reliably reach a designated level of throughput. This is useful in data protection scenarios where either:

- a dedicated link is used for data protection traffic.
- a known percentage of a given link can be fully consumed by data protection traffic.

For example, if an EMC SRDF/A replication deployment is using peer 6050 Steelhead appliances that are connected to a dedicated OC-1 link (50 Mbps), then you can create a MX-TCP class of 50 Mbps on each Steelhead appliance. In this example, SRDF/A uses port 1748 for data transfers.

On both the client and server-side Steelhead appliances, enter the following commands:

```
qos shaping interface wan0_0 rate 50000
qos shaping interface wan0_0 enable
qos classification enable
qos classification class add class-name "blast" priority realtime min-pct 99.0000000 link-share
100.0000000 upper-limit-pct 100.0000000 queue-type mxtcp queue-length 100 parent "root"
qos classification rule add class-name "blast" traffic-type optimized destination port 1748 rulenum
1
qos classification rule add class-name "blast" traffic-type optimized source port 1748 rulenum 1
write mem
restart
```

If you cannot allocate a given amount of bandwidth for data protection traffic, but you still require high bandwidth, enable High-Speed TCP (HS-TCP) on peer Steelhead appliances.

Note: To configure MX-TCP, you must enable advanced outbound QoS on the Steelhead appliance. More details, see [“MX-TCP” on page 93](#).

For details on fat pipes, see [“Underutilized Fat Pipes” on page 445](#). For details on MX-TCP, see [“MX-TCP” on page 93](#).

For more information about MX-TCP as a transport streaming lining mode, see [“Overview of Transport Streamlining” on page 9](#).

The Steelhead Appliance WAN Buffer Settings

In all data protection scenarios, set the Steelhead appliance WAN buffers to at least 2 x BDP. For example, if NetApp SnapMirror traffic is using a dedicated OC-1 link (50 Mbps) with 30 ms of latency (60 ms round-trip time) between sites, then set the Steelhead appliance WAN-side buffers to:

$$2 \times \text{BDP} = 2 \times 50 \text{ Mb/s} \times 1,000,000 \text{ b/Mb} \times 60 \text{ ms} \times (1/1000) \text{ s/ms} \times (1/8) \text{ Bytes/bit} = 750,000 \text{ Bytes}$$

On all Steelhead appliances in this environment that send or receive the data protection traffic, enter the following commands:

```
protocol connection wan send def-buf-size 750000
protocol connection wan receive def-buf-size 750000
write mem
restart
```

The Router WAN Buffer Settings

In environments where a small number of connections are transmitting high-throughput data flows, you must increase the WAN-side queues on the router to the BDP.

For example, consider an OC-1 link (50 Mbps) with 60 ms latency (RTT):

$$\text{BDP} = 50 \text{ Mbps} * 1,000,000 \text{ b/Mb} * 60 \text{ ms} * (1/1000) \text{ s/ms} * (1/8) \text{ Bytes/bit} * (1/1500) \text{ Bytes/packet} \\ = 250 \text{ Packets}$$

On the Cisco router, enter the following hold-queue interface configuration command:

```
hold-queue 250 out
```

You do not need to increase the router setting when using MX-TCP because MX-TCP moves bottleneck queuing onto the Steelhead appliance. This feature allows WAN traffic to enter the network at a constant rate, eliminating the need for excess buffering on router interfaces.

Common Data Protection Deployments

This section describes common data protection deployments. This section includes the following topics:

- [“Remote Office, Branch Office Backups” on page 349](#)
- [“Network Attached Storage Replication” on page 349](#)
- [“Storage Area Network Replication” on page 350](#)

Remote Office, Branch Office Backups

The remote office, branch office (ROBO) data protection deployment is characterized by one or more small branch office locations, each of which backs up file data from one or more file servers, PCs, and laptops to a central data center. Common applications include Veritas NetBackup, EMC Legato, CommVault Simpana, Sun StorageTek, and backups performed over standard protocols like CIFS and FTP.

In these deployments, WAN links are relatively small, commonly ranging from 512 Kbps on the low end to 10 Mbps on the high end. Also distinct from data center-to-data center replication scenarios where dedicated Steelhead appliances are typically used exclusively for replication, ROBO backup procedures commonly use the same branch office Steelhead appliances that are used to accelerate other applications, like CIFS and MAPI. For both of these reasons, ROBO backups commonly require relatively larger levels of WAN bandwidth reduction.

In the Performance page ([Figure 16-2](#)), enter the initial configuration of the peer Steelhead appliances as follows:

- **Set the Adaptive Streamlining mode to Default** - Due to limited WAN bandwidth in these deployments, it is important to maximize WAN data reduction. The default setting uses disk-based RiOS SDR to provide maximum data reduction. File backup workloads typically result in sequential disk access which works well for disk-based RiOS SDR.
- **Set the Compression Level to 6** - Start with aggressive compression to minimize WAN bandwidth.
- **Enable Multicore balancing** - This option allows the Steelhead appliance to use all CPU cores even when there are a small number of connections. Small connection counts can occur if backups are performed nightly, when little to no additional traffic is generated.

Network Attached Storage Replication

Network attached storage (NAS) data protection deployment sends primary file data over the WAN to online replicas. Common applications include NetApp SnapMirror, EMC VNX Replicator, and VNX Celerra Replicator.

For details on EMC qualification matrix for Riverbed Technology, see the Riverbed Knowledge Base article *Deploying Steelhead Appliances with EMC Storage*, at <https://supportkb.riverbed.com/support/index?page=content&id=s13363>.

In NAS replication deployments, WAN links are typically large, ranging from T3 (45 Mbps) to OC-48 (2.5 GB). Often NAS replication solutions require dedicated links used exclusively by the NAS replication solution.

As a best practice for high-speed NAS replication solutions, use Steelhead appliances that are dedicated to only optimizing high-speed NAS replication workloads, and that do not optimize large amounts of general application or end-user traffic. This benefits you for the following reasons:

- Increase both the level and predictability of performance delivered by Steelhead appliances, leading to consistent delivery of recovery point and time objectives (RPO/RTO).
- With separate Steelhead appliances, the large data sets commonly associated with high-speed replication do not compete for Steelhead appliance data store resources with other user-based traffic, and the reverse.
- You can optimally tune separate Steelhead appliances for their respective workloads.

Disable any data compression applied on the storage device so that data enters the Steelhead appliance in its raw form. Disabling data compression enables the Steelhead appliance to perform additional bandwidth reduction using RiOS SDR.

In the Performance page ([Figure 16-2](#)), enter the initial configuration of the peer Steelhead appliances as follows:

- **Set the Compression Level to 1** - Higher compression levels produce additional gains in WAN-side bandwidth reduction, but often at a large cost to the CPU resources, which ultimately throttles LAN-side throughput.
- **Enable Multicore Balancing** - Often there are a small number of connections made between storage devices. This option enables the optimization services to balance their processing across all CPU cores.
- **Enable MX-TCP or HS-TCP** - If there is a dedicated WAN-link for the NAS replication traffic or if you know how much bandwidth on a shared link can be allocated to the data transfer, create an MX-TCP class covering the data traffic. If not, enable HS-TCP. If HS-TCP is enabled, increase the router queue length to the BDP. Configure MX-TCP on the QoS Classification page.
- Set the Steelhead appliance WAN buffers to 2 x BDP. This option allows the Steelhead appliances to buffer enough data to continue accepting data from the LAN—even in cases of WAN packet loss.

In cases where WAN links exhibit high-packet loss, you might need to increase the Steelhead appliance WAN buffers higher than 2 x the BDP for optimal throughput.

Storage Area Network Replication

Storage area network (SAN) data protection deployment includes SAN replication products such as EMC Symmetrix Remote Data Facility/Asynchronous (SRDF/A), IBM Global Mirror/IBM Global Mirror, and Hitachi Universal Replicator, including full and incremental backups of databases like Oracle and Exchange.

For more information about SAN replications, the chapter on [“Storage Area Network Replication” on page 357](#).

Designing for Scalability and High Availability

Scalability and high availability are often required in data protection deployments. This section describes the design of data protection solutions which address both requirements. This section includes the following topics:

- [“Overview of N+M Architecture” on page 351](#)
- [“Using MX-TCP in N+M Deployments” on page 351](#)

For more information about high availability, see [“Multiple WAN Router Deployments” on page 192](#).

Overview of N+M Architecture

The most cost-effective way to provide scalability and high availability is by using a N+M Steelhead appliance architecture, or an N+M Deployment. In an N+M architecture, N represents the minimum number of Steelhead appliances that are required to process the total amount of traffic from site to site. M represents the number of additional Steelhead appliances needed to provide a desired amount of redundancy. For example, a common requirement is to maintain availability in the presence of a single failure. In this case, you can use a N+1 Steelhead appliance deployment architecture.

Using MX-TCP in N+M Deployments

This section describes how to use MX-TCP in N+M deployments. This section includes the following topics:

- [“The Interceptor Appliance and N+M Active and Backup Deployment” on page 352](#)
- [“The Interceptor Appliance and Pass-Through Connection Blocking Rules” on page 353](#)

MX-TCP is typically used in data protection deployments when all or part of the WAN bandwidth is dedicated to the data transfers. When using MX-TCP with multiple Steelhead appliances, MX-TCP settings are set on each Steelhead appliance so that the collection of Steelhead appliances uses the available WAN bandwidth.

For details, see [“QoS in Multiple Steelhead Appliance Deployments” on page 104](#), and [“MX-TCP” on page 93](#).

In an N+M deployment, the following options effect how to configure MX-TCP:

- **All Active, or N+M Active** - All N+M Steelhead appliances participate in optimizing the data transfer. Configure MX-TCP on each Steelhead appliance to use $1/(N+M)$ th of the total available WAN bandwidth. For example, in a 2+1 All Active deployment, configure MX-TCP on each Steelhead appliance to use one-third of the available bandwidth. Less WAN bandwidth is used when one or more Steelhead appliances are offline. For example, in a 2+1 All Active deployment with one Steelhead appliance offline, two-third of the allocated WAN bandwidth is used by the Steelhead appliances that remain online.
- **Active and Backup, or N Primary + M Backup** - Exactly N Steelhead appliances participate in optimizing the data transfer. Configure MX-TCP on each Steelhead appliance to use $1/N$ th of the total available WAN bandwidth. If one or more active Steelhead appliances are offline, backup Steelhead appliances are used to keep the WAN fully utilized. For example, in a 2+1 Active and Backup deployment, configure MX-TCP on each Steelhead appliance to use one-half of the available bandwidth. If one active Steelhead appliance is offline, the backup Steelhead appliance participates in optimizing the data transfer, keeping the WAN fully utilized.

For details on how to configure an Active and Backup deployment using the Interceptor appliance, see [“The Interceptor Appliance and N+M Active and Backup Deployment” on page 352](#)).

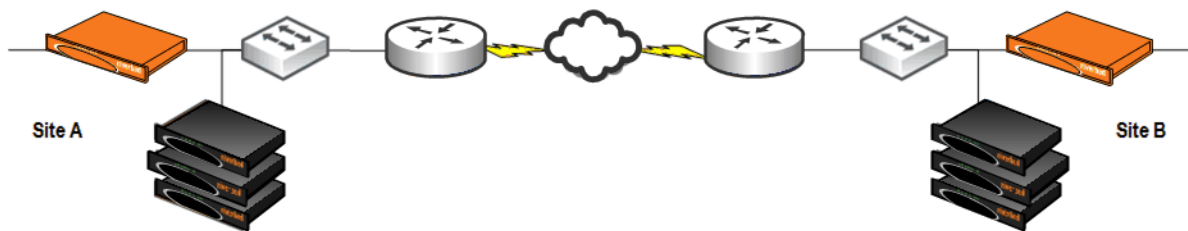
The Interceptor Appliance and N+M Active and Backup Deployment

When configuring the Interceptor appliance for an N+M Active and Backup deployment using the Interceptor appliance, load-balance rules are defined which carry out the following actions:

- Balance load across the primary Steelhead appliances
- Use backup Steelhead appliance in the event of a failure

Figure 16-3 shows a 2+1 Active and Backup deployment.

Figure 16-3. Interceptor Appliance N+M



In each site there is an Interceptor appliance and three Steelhead appliances: two are primary and one is the backup. Connections are established from Site A to Site B, and there are four hosts (not depicted) at each site that process equal amounts of data. The following is a list IP addresses for the hosts and Steelhead appliances at Site A:

- Hosts 1-4: 10.30.50.11 - 10.30.50.14
- Primary Steelhead 1: 10.30.50.15
- Primary Steelhead 2: 10.30.50.16
- Backup Steelhead: 10.30.50.17

The following load-balance rules are used on each Interceptor appliance to evenly split the connections established from the four hosts at Site A across the two primary Steelhead appliances (odd-numbered hosts are redirected to primary Steelhead 1, and even-numbered hosts are redirected to primary Steelhead 2).

```
load balance rule redirect addrs 10.30.50.15 src 10.30.50.11/32
load balance rule redirect addrs 10.30.50.16 src 10.30.50.12/32
load balance rule redirect addrs 10.30.50.15 src 10.30.50.13/32
load balance rule redirect addrs 10.30.50.16 src 10.30.50.14/32
```

The following load-balance rules allow the Interceptor appliance to use the backup Steelhead appliance in case either of the primary Steelhead appliances fails:

```
load balance rule redirect addrs 10.30.50.17 src 10.30.50.11/32
load balance rule redirect addrs 10.30.50.17 src 10.30.50.12/32
load balance rule redirect addrs 10.30.50.17 src 10.30.50.13/32
load balance rule redirect addrs 10.30.50.17 src 10.30.50.14/32
```

The same configuration would be used for the Interceptor appliance at Site B, using instead of the IP addresses for the Steelhead appliances in site B.

The Interceptor Appliance and Pass-Through Connection Blocking Rules

In some data protection deployments, it is important to prevent backup and replication connections from being established as un-optimized, or pass-through, connections. These un-optimized connections can have a negative impact on meeting LAN and WAN throughput objectives. Interceptor v2.0.3 or later supports *Pass-through Connection Blocking Rules*. This feature adds a set of rules that can break existing pass-through connections and prevent formation of new ones.

For example, you can create a pass-through blocking rule for port 1748, connect to the Interceptor CLI and enter the following command:

```
in-path passthrough rule block port start 1748 end 1748
```

For details, see the *Interceptor Appliance User's Guide* and *Riverbed Command-Line Interface Reference Manual*.

SnapMirror Optimization

The Steelhead appliance can optimize SnapMirror traffic by removing huge amounts of repetitive data from the WAN and by compressing the data that is sent across during mirror initialization and updates. SnapMirror optimization support is for environments using NetApp Data ONTAP v7 or Data ONTAP v8 operating in 7-Mode.

The two varieties of SnapMirror are volume-based and qtree-based. SnapMirror replicates data from one volume or qtree (the source) to another volume or qtree (the mirror). SnapMirror periodically updates the mirror to reflect incremental changes to the source. The result of this process is an online, read-only volume (the mirror), that contains the same data as the source volume at the time of the most recent update.

You can use the information on the mirror to:

- provide quick access to data in the event of a disaster that makes the source volume or qtree unavailable. The secondary copy is nearly identical to the primary copy; every snapshot on the primary copy also exists on the backup copy. You can schedule updates as frequently as every minute.
- update the source to recover from disaster, data corruption (mirror qtrees only), or user error.
- archive the data to tape.
- balance resource loads.
- back up or distribute the data to remote sites.

SnapMirror optimization in RiOS v8.5 or later allows you to:

- customize SnapMirror traffic optimization on per-volume or qtree basis. You can apply desired optimization algorithms (SDR-Default, LZ-only, and None) for different data types that reside on targeted volumes or qtrees.
- apply QoS traffic shaping policies on per-volume or qtree basis. You can assign mappings by filer and volume name to one of five volume priorities. Using advanced QoS, you can assign a service class and DSCP value to each volume priority when creating a rule for SnapMirror traffic. Multipath operations are not supported.
- collect and chart SnapMirror statistics such as the total LAN and WAN bytes in and out, throughput, the data reduction at the filer, and volume/qtrees granularity.

To benefit from the improved SnapMirror optimization, both Steelhead appliances must be running RiOS v8.5 or later. SnapMirror optimization is disabled by default.

The following example shows a company with a NetApp filer that is replicating four volumes from New York to San Francisco. These four volumes contain the following data:

- **Volume 1** - contains archival data of the previous year. The data continues to reside on the primary storage because it is used regularly by the analytics department. However, to preserve server space, the data is stored in compressed format.
- **Volume 2** - contains graphics and videos. The data is encrypted or not compressible.
- **Volume 3** - contains an MS Exchange data store of the company users email mailboxes.
- **Volume 4** - contains lab data that for historical reasons is produced and stored in plain 7-bit ACSII format, but doesn't have repeatable patterns, as usual text would.

Because the data composition varies by volume, to best use the Steelhead appliance optimization resources, and to accomplish the best result on the data optimization, you can apply different optimization methods for each volume.

Data in Volume 1 and Volume 2 is neither compressible nor dedupable. If you enable SDR, you have a higher CPU optimization on the Steelhead appliances and you do not achieve the best use of the SDR data store. The SDR data store can provide higher data reduction to the rest of the data. However, the Steelhead appliance can add value by sending the data using MX-TCP protocol. MX-TCP protocol enables the most of available bandwidth. Traditional TCP cannot accomplish this because of packet loss, high latency, or some combination of these and other impairments.

Data in Volume 3 benefits from SDR because it contains repeatable patterns in the data stream that can both be compressed and replaced with references.

Data in Volume 4 benefits from conventional compression or LZ-only type of Steelhead optimization.

Different volumes can have different change rates and therefore have replication service level agreements (SLAs) or recovery point and/or time objectives (RPO/RTO): for example, four-hour RTO for MS Exchange and 24-hour RTO for images and video. To meet varied requirements, you can apply different QoS policies for those volumes to make increase priority of MS Exchange (Volume 3) data or image data (Volume 2).

For more information about the configuring SnapMirror on the Steelhead appliance, see the *Steelhead Appliance Management Console User's Guide*. For more information about SnapMirror and QoS, see ["Configuring QoS for SnapMirror" on page 145](#).

Troubleshooting and Fine-Tuning

If your data protection deployment is not meeting performance targets after configuring the Steelhead appliances using the methods described in this chapter, examine the following system components for potential bottlenecks:

- **Application Servers** - Are the server and client fast enough? To perform a LAN baseline check, put the Steelhead appliances in bypass mode and connect the servers directly through a high-bandwidth network with zero latency to see how fast they are. Time permitting, you might want to do this LAN baselining before introducing the Steelhead appliances into the test environment.
- **LAN-side Network** - Make sure that there are no issues with the LAN-side network between the Steelhead appliances and any data protection hosts. In particular, on the LAN, there should be no packet loss, and the round trip latency between the Steelhead appliances and hosts should be less than one millisecond for the fastest possible throughput. Interface errors, especially those related to Ethernet duplex negotiation, are a leading factors of LAN-side network issues.

- **WAN-side Network** - Use MX-TCP to overcome any WAN-side packet loss caused by deficient links or undersized router interface queues. If the WAN bandwidth is being fully utilized during optimized data transfers, then the WAN is the bottleneck. If the WAN link is not fully utilized, options like RiOS SDR-A or SDR-M can increase the LAN-side throughput.
- **CPU** - Check the CPU reports to see if the CPU cores are the bottleneck. If some cores are busy but some are not, enable multicore load balancing. If you enable multicore load balancing and all cores are fully utilized, you might require a larger model Steelhead appliance.
- **Disk** - You can use disk-related metrics to determine that the disk is the bottleneck for higher levels of throughput. Always assess these metrics relative to empirical application performance. Even if they indicate heavy disk utilization, it does not necessarily mean that the disk is the bottleneck. In cases where the disk is the bottleneck, then you can adjust the adaptive data streamlining settings progressively upward to either SDR-A, SDR-M or, finally, compression-only. In some cases, you might need to upgrade to a higher model Steelhead appliance. Consult with your Riverbed Sales or Professional Services representative.
- **Data Store Disk Load** - If the RiOS Data Store Read Efficiency report, accessible from the Management Console, shows that read efficiency falls below 50% consistently, this might indicate that the disk is the bottleneck.

Third-Party Interoperability

Riverbed optimizes data protection utilities from many storage vendors, including but not limited to the following:

- EMC
For details on EMC qualification matrix for Riverbed Technology, see the Riverbed Knowledge Base article *Deploying Steelhead Appliances with EMC Storage*, at <https://supportkb.riverbed.com/support/index?page=content&id=s13363>.
- NetApp
- HP
- Hitachi Data Systems (HDS)
- IBM
- Dell
- Brocade
- QLogic
- Symantec/Veritas
- Microsoft
- Commvault
- DoubleTake
- CA
- Compellent
- 3Par
- BlueArc

For additional information go to the following Web sites:

- <http://riverbed.com>
- <http://support.riverbed.com>

Alternatively, you can consult with your authorized Riverbed Solutions Provider.

CHAPTER 17 Storage Area Network Replication

Storage area network (SAN) data protection deployment includes SAN replication products such as EMC Symmetrix Remote Data Facility / Asynchronous (SRDF / A), IBM Global Mirror, and Hitachi Universal Replicator, including full and incremental backups of databases like Oracle and Exchange. You can use TCIP/IP or FCIP connectivity to transfer SAN Replication traffic. This chapter includes the following sections:

- [“Overview of SAN Replication” on page 357](#)
- [“Storage Optimization Modules” on page 358](#)
- [“Best Practices for SAN Replication Using TCP/IP” on page 366](#)
- [“Best Practices for SAN Replication Using FCIP” on page 367](#)

SAN replication is one of the common deployments for data protection. For information about the other options, see [“Common Data Protection Deployments” on page 349](#).

Overview of SAN Replication

In SAN replication deployments, WAN links are typically large, often ranging from T3 (45 Mbps) to OC-48 (2.5 Gbps) or more. Often SAN replication solutions require dedicated links used exclusively by the SAN replication solution.

As a best practice for high-speed SAN replication solutions, use Steelhead appliances that are dedicated to only optimizing high-speed SAN replication workloads, and that do not optimize large amounts of general application or end-user traffic. This benefits you for the following reasons:

- Increase both the level and predictability of performance delivered by Steelhead appliances, leading to consistent delivery of recovery point and time objectives (RPO/RTO).
- With separate Steelhead appliances, the large data sets commonly associated with high-speed replication do not compete for Steelhead appliance data store resources with other user-based traffic, and the reverse.
- You can optimally tune separate Steelhead appliances for their respective workloads.

Disable any data compression on the SAN array (for example, EMC Symmetrix Gigabit Ethernet connectivity) and on the FCIP or iFCP gateways (for example, Cisco MDS, Brocade, QLogic, and McData Eclipse), so the data enters the Steelhead appliance in raw form. Disabling data compression allows the Steelhead appliances the opportunity to perform additional bandwidth reduction using RiOS SDR.

Use dedicated Steelhead appliances of the same model for this type of data protection scenario. Consult with your SAN vendor's customer service representative for best practice configuration of their arrays for use with Steelhead appliances.

For details on EMC qualification matrix for Riverbed Technology, see the Riverbed Knowledge Base article *Deploying Steelhead Appliances with EMC Storage*, at <https://supportkb.riverbed.com/support/index?page=content&id=s13363>.

Storage Optimization Modules

This section describes the storage optimization module options. This section includes the following topics:

- [“FCIP Optimization Module” on page 358](#)
- [“SRDF Optimization Module” on page 361](#)

RiOS v6.0.1 or later includes storage optimization modules for the FCIP and SRDF protocols. These modules provide enhanced data reduction capabilities. The modules use explicit knowledge of where protocol headers appear in the storage replication data stream to separate out headers from the payload data that was written to storage. In absence of a module, these headers represent an interruption to the network stream, reducing the ability of RiOS SDR to match on large, contiguous data patterns.

The modules must be configured based on the types of storage replication traffic present in the network environment. The following sections describe these options and when they would be applied.

FCIP Optimization Module

This section describes the storage optimization for FCIP and how to configure it. This section includes the following topics:

- [“Configuring Base FCIP Module” on page 359](#)
- [“Configuring FCIP Module Rules” on page 360](#)

The module for FCIP is appropriate for environments using storage technology that originates traffic as fibre channel (FC) and then uses a Cisco MDS or Brocade FCIP gateway to convert the FC traffic to TCP for WAN transport.

For details on storage technologies that originate traffic via FC see, [“Storage Area Network Replication” on page 357](#). For configuration best-practice details for Cisco MDS and Brocade FCIP deployments, see [“Best Practices for SAN Replication Using FCIP” on page 367](#).

Note: Environments with SRDF traffic originated via Symmetrix FC ports (RF ports) only require configuration of the RiOS FCIP module. The SRDF module only applies to traffic originated through Symmetrix Gigabit Ethernet ports (RE ports). For details, see [“SRDF Optimization Module” on page 361](#).

All configuration for FCIP must be applied on the Steelhead appliance closest to the FCIP gateway that opens the FCIP TCP connection by sending the initial SYN packet. If you are unsure which gateway initiates the SYN in your environment, Riverbed recommends that you apply the module configuration to the Steelhead appliances on both ends of the WAN.

Configuring Base FCIP Module

By default, the FCIP module is disabled. When only the base FCIP module has been enabled, all traffic on the well-known FCIP TCP destination ports 3225, 3226, 3227, and 3228 are directed through the module for enhanced FCIP header isolation. In most environments, no further FCIP module configuration is required beyond enabling the base module.

To enable the base FCIP module

1. Connect to the Steelhead CLI and enter the following command:

```
protocol fcip <enable/disable>
```

2. If an environment uses one or more non-standard TCP ports for FCIP traffic, the module can be configured to handle traffic on additional ports by entering the following command:

```
protocol fcip ports <port-list>
```

Where **<port-list>** is a comma-separated list of TCP ports. Prefix this command with **no** to remove one or more TCP ports from the list of those currently directed to the FCIP module.

You can check whether the module is currently enabled or disabled, and on which TCP ports the module is looking for FCIP traffic.

To show current base FCIP module settings

- Connect to the Steelhead CLI and enter the following command:

```
show protocol fcip settings
```

The Current Connections report shows optimized connections with the *App* label for each connection shown as *FCIP*, if the base FCIP module is enabled and connections are established. If the report shows a connection's *App* as *TCP*, the module is not used and the configuration must be checked.

To observe the current base FCIP module connections

- Connect to the Steelhead CLI and enter the following command:

```
show connections
```

T	Source	Destination	App	Rdn	Since
O	10.12.254.2	4261 10.12.254.34	3225 FCIP	18%	2010/03/09 18:50:02
O	10.12.254.2	4262 10.12.254.34	3226 FCIP	86%	2010/03/09 18:50:02
O	10.12.254.142	4315 10.12.254.234	3225 FCIP	2%	2010/03/09 18:50:02
O	10.12.254.142	4316 10.12.254.234	3226 FCIP	86%	2010/03/09 18:50:02

Configuring FCIP Module Rules

An environment that has RF-originated SRDF traffic between VMAX arrays requires additional configuration beyond enabling the FCIP base module. Specifically, the SRDF protocol implementation used to replicate between two VMAX arrays uses an additional *Data Integrity Field* (DIF) header, which further interrupts the data stream. For Open Systems environments (such as Windows and UNIX/Linux), the DIF header is injected into the data stream after every 512 bytes of storage data. For IBM iSeries (AS/400) environments, the DIF header is injected after every 520 bytes. Do not add a module rule isolating DIF headers in mainframe environments, as SRDF environments that replicate mainframe traffic do not currently include DIF headers.

Note: FCIP module rules are only required for VMAX-to-VMAX traffic.

If your environment includes RF-originated SRDF traffic between VMAX arrays, the module can be configured to look for DIF headers.

To configure the FCIP module to look for DIF headers in the FCIP data stream

- Connect to the Steelhead CLI and enter the following command:

```
protocol fcip rule src-ip <IP address> dst-ip <IP address> dif <enable/disable> dif-blocksize  
<number of bytes>
```

For example, if the only FCIP traffic in your environment is RF-originated SRDF between VMAX arrays, you can allow for isolation of DIF headers on all FCIP traffic by modifying the default rule as follows:

```
protocol fcip rule src-ip 0.0.0.0 dst-ip 0.0.0.0 dif enable
```

Environments that have a mix of VMAX-to-VMAX RF-originated SRDF traffic along with other FCIP traffic require additional configuration, because Steelhead appliances must be informed where DIF headers are expected. This configuration is made based on IP addresses of the FCIP gateways. In such a mixed environment, SAN zoning needs to be applied to ensure that DIF and non-DIF traffic are not carried within the same FCIP tunnel.

Assume your environment consists mostly of regular, non-DIF FCIP traffic but also some RF-originated SRDF between a pair of VMAX arrays. Assume a pair of FCIP gateways are configured with a tunnel to carry the traffic between these VMAX arrays, and that the source IP address of the tunnel is 10.0.0.1 and destination IP is 10.5.5.1. The pre-existing default rule tells the module not to expect DIF headers on FCIP traffic. This setting allows for correct handling of the all the non-VMAX FCIP. To obtain the desired configuration, enter the following command to override the default behavior and perform DIF header isolation on the FCIP tunnel carrying the VMAX-to-VMAX SRDF traffic:

```
protocol fcip rule src-ip 10.0.0.1 dst-ip 10.5.5.1 dif enable
```

When configured, the FCIP module looks for a DIF header after every 512 bytes of storage data, which is typical for an Open Systems environment. If your environment uses IBM iSeries (AS/400) hosts, use the **dif-blocksize** to inform the module to look for a DIF header after every 520 bytes of storage data. Enter the following command to modify the default rule to look for DIF headers on all FCIP traffic in a VMAX based, IBM iSeries (AS/400) environment:

```
protocol fcip rule src-ip 0.0.0.0 dst-ip 0.0.0.0 dif enable dif-blocksize 520
```

To observe the current base FCIP module connections

- Connect to the Steelhead CLI and enter the following command:

```
show protocol fcip rules
```

You can display each rule currently configured, whether DIF header isolation is enabled or disabled for that rule, and how much storage data is expected before each DIF header in traffic matching that rule.

SRDF Optimization Module

This section describes the storage optimization for SRDF and how to configure it. This section includes the following topics:

- [“Configuring the Base SRDF Module” on page 361](#)
- [“Detecting Symmetrix VMAX Microcode” on page 362](#)
- [“Configuring SRDF Module Rules” on page 363](#)
- [“Configuring SRDF Selective Optimization” on page 364](#)
- [“Viewing SRDF Reports” on page 365](#)

The module for SRDF is appropriate for environments using EMC's Symmetrix Remote Data Facility (SRDF) with DMX and VMAX storage arrays when the traffic is originated directly from Gigabit Ethernet ports on the arrays (also referred to as *RE* ports). When in this configuration, the SRDF traffic appears on the network immediately as TCP. The SRDF protocol injects headers into the data stream; these headers interrupt the continuity of Steelhead Data Reduction (SDR). The SRDF module removes these headers from the data stream before performing data reduction, and then re-injects them before sending data to the receiving EMC Symmetrix. In addition, the SRDF module automatically disables native EMC SRDF compression for SRDF transfers, which prevents the need for a BIN file change to disable compression. In the event of a Steelhead appliance or network failure, the Symmetrix arrays fall back on native compression instead of transmitting at uncompressed bandwidth rates.

Note: Environments with SRDF traffic originated through Symmetrix fibre channel ports (*RF* ports) require configuration of the RiOS FCIP module, not the SRDF module. For details on RF ports, see [“FCIP Optimization Module” on page 358](#).

All configuration for SRDF must be applied on the Steelhead appliance closest to the Symmetrix array that opens the SRDF TCP connection by sending the initial SYN packet. If you are unsure which array initiates the SYN in your environment, Riverbed recommends that you apply module configuration to the Steelhead appliances on both ends of the WAN.

Configuring the Base SRDF Module

By default, the SRDF module is disabled. When only the base SRDF module has been enabled, all traffic on the well-known SRDF TCP destination port 1748 is directed through the module for enhanced header isolation. In most environments using SRDF only between DMX arrays or VMAX-to-DMX, no further SRDF module configuration is required beyond enabling the base module.

To enable the base SRDF module

1. Connect to the Steelhead CLI and enter the following command:

```
protocol srdf <enable/disable>
```

2. If an environment used one or more non-standard TCP ports for RE-originated SRDF traffic, the module can be configured to handle traffic on additional ports by entering the following command:

```
protocol srdf ports <port-list>
```

Where <port-list> is a comma-separated list of TCP ports. Prefix this command with **no** to remove one or more TCP ports from the list of those currently directed to the SRDF module.

You can see whether the module is currently enabled or disabled, and on which TCP ports the module is looking for SRDF traffic.

To observe current base SRDF module settings

- Connect to the Steelhead CLI and enter the following command:

```
show protocol srdf settings
```

The Current Connections report shows optimized connections with the *App* label for each connection shown as *SRDF*, if the base SRDF module is enabled and connections are established. If the report shows a connection's *App* as *TCP*, the module is not used and the configuration must be checked.

To observe the current base SRDF module connections

- Connect to the Steelhead CLI and enter the following command:

```
show connections
```

T	Source	Destination	App	Rdn	Since
O	10.12.254.80	4249 10.12.254.102	1748 SRDF	82%	2010/03/09 16:35:40
O	10.12.254.80	4303 10.12.254.202	1748 SRDF	83%	2010/03/09 16:35:40
O	10.12.254.180	4250 10.12.254.102	1748 SRDF	85%	2010/03/09 16:35:40
O	10.12.254.180	4304 10.12.254.202	1748 SRDF	86%	2010/03/09 16:35:40

Detecting Symmetrix VMAX Microcode

For Symmetrix VMAX running Enginuity microcode levels newer than 5874, you do not need to configure SRDF module rules (for details, see [“Configuring SRDF Selective Optimization” on page 364](#)). For Symmetrix VMAX running Enginuity level 5874 or older, configure SRDF module rules as described in the following section.

To detect Symmetrix microcode level for Open Systems-connected Symmetrix, use the **symcfg** command in EMC's Solutions Enabler software. Solutions Enabler is EMC software is typically used for managing Symmetrix storage arrays. The following is example output:

```
# symcfg -sid 000194900363 list -v
Symmetrix ID: 000194900363
Time Zone : PST
Product Model : VMAX-1SE
Symmetrix ID : 000194900363

Microcode Version (Number) : 5875 (16F30000)
Microcode Registered Build : 0
Microcode Date : 11.22.2010

Microcode Patch Date : 11.22.2010
Microcode Patch Level : 122
```

Configuring SRDF Module Rules

An environment that has RE-originated SRDF traffic between VMAX arrays requires additional configuration beyond enabling the base module. Specifically, the SRDF protocol implementation used to replicate between two VMAX arrays employs an additional *Data Integrity Field* (DIF) header, which further interrupts the data stream. For Open Systems environments (such as Windows and UNIX/Linux), the DIF header is injected into the data stream after every 512 bytes of storage data. For IBM iSeries (AS/400) environments the DIF header is injected after every 520 bytes. Do not add a module rule isolating DIF headers in mainframe environments, as SRDF environments that replicate mainframe traffic do not currently include DIF headers.

Note: SRDF module rules are only required for VMAX-to-VMAX traffic.

If your environment includes RE-originated SRDF traffic between VMAX arrays, the module can be configured to look for DIF headers.

To configure the SRDF module to look for DIF headers

- Connect to the Steelhead CLI and enter the following command:

```
(config) # protocol srdf rule src-ip <IP address> dst-ip <IP address> dif <enable/disable> dif-blocksize <number of bytes>
```

For example, if the only RE-originated SRDF traffic in your environment is between VMAX arrays, you can allow for isolation of DIF headers on all SRDF traffic by modifying the default rule as follows:

```
(config) # protocol srdf rule src-ip 0.0.0.0 dst-ip 0.0.0.0 dif enable
```

Environments that have a mix of VMAX-to-VMAX and DMX-based SRDF traffic require additional configuration, because Steelhead appliances must be informed where DIF headers are expected. This configuration is made based on RE port IP addresses.

Assume your environment contained RE-originated SRDF traffic mostly between DMX arrays but also some between a pair of VMAX arrays. Assume the VMAX array in the primary location had RE ports of IP addresses 10.0.0.1 and 10.0.0.2 and the VMAX array in the secondary location had RE ports of IP addresses 10.5.5.1 and 10.5.5.2. The pre-existing default rule tells the module not to expect DIF headers on all RE-originated SRDF traffic. This allows for correct handling of the main DMX-based SRDF traffic. To obtain the desired configuration, enter the following commands to override the default behavior and perform DIF header isolation on the VMAX SRDF connections:

```
(config) # protocol srdf rule src-ip 10.0.0.1 dst-ip 10.5.5.1 dif enable
(config) # protocol srdf rule src-ip 10.0.0.1 dst-ip 10.5.5.2 dif enable
(config) # protocol srdf rule src-ip 10.0.0.2 dst-ip 10.5.5.1 dif enable
(config) # protocol srdf rule src-ip 10.0.0.2 dst-ip 10.5.5.2 dif enable
```

When configured, the SRDF module looks for a DIF header after every 512 bytes of storage data, which is typical for an Open Systems environment. If your environment uses IBM iSeries (AS/400) hosts, rules that use the **dif-blocksize** to inform the module to look for a DIF header after every 520 bytes of storage data. Enter the following command to modify the default rule to look for DIF headers on all SRDF traffic in a

VMAX based, IBM iSeries (AS/400) environment:

```
(config) # protocol srdf rule src-ip 0.0.0.0 dst-ip 0.0.0.0 dif enable dif-blocksize 520
```

To observe the current SRDF rule settings

- Connect to the Steelhead CLI and enter the following command:

```
show protocol srdf rules
```

This shows each rule currently configured, whether DIF header isolation is enabled or disabled for that rule, and how much storage data is expected before each DIF header in traffic matching that rule.

Configuring SRDF Selective Optimization

RiOS v6.1.2 or later feature *selective optimization*. Selective optimization is the ability to provide different types of optimization to different RDF Groups, and allows you to tune for the best optimization setting for each RDF group to maximize the Steelhead appliance usability. Selective optimization also depends on Symmetrix VMAX Engenuity microcode levels newer than 5874.

Consider an example with three types of data:

- Oracle logs (RDF group 1)
- Encrypted check images (RDF group 2)
- Virtual machine images (RDF group 3)

In this example, assign LZ-only compression to the Oracle logs, no optimization to the encrypted check images, and default SDR to the virtual machine images. To assign these levels of optimization, configure the Steelhead appliance to associate specific RE port IP addresses with specific Symmetrix arrays, and then assign rules to specific RDF groups for different optimization policies.

To configure the Steelhead appliance to associate RE ports with a specific Symmetrix

1. Connect to the Steelhead appliance CLI and enter the following command:

```
(config) # protocol srdf symm id <SYMMID> address <IP address>
```

The Symmetrix ID is an alphanumeric string that can contain hyphens and underscores (for example, a standard Symmetrix serial number is 000194900363). Do not use spaces or special characters.

2. Add a rule to affect traffic coming from the RE ports associated with this SYMMID:

```
(config) # protocol srdf symm id <SYMMID> rdf_group <RDF_GROUP> optimization <OPT POLICY>
[description *]
```

OPT_POLICY is one of: none, lz-only, or sdr-default.

RDF_GROUP in RiOS is specified as a decimal number; however some EMC utilities report RDF Group Numbers in hexadecimal.

When you have Symmetrix arrays serving Open Systems hosts, and you are using EMC Solutions Enabler, RDF group numbers are reported in decimal—ranging from 1 to 255. By default, this is how RDF_GROUP is entered in RiOS and shown in reports. For mainframe-attached Symmetrix arrays, tools report RDF group numbers in hexadecimal, starting from 0. Use the following command for Steelhead appliances serving only mainframe-attached Symmetrix arrays, and you want the representation of RDF_GROUP to be in the range 0 to 254:

```
(config) # protocol srdf symm id base_rdf_group 0
```

In the example of three RDF groups (assuming a SYMMID of 123), enter the following commands:

```
(config) # protocol srdf symm id 123 rdf group 1 optimization lz-only description Oracle1_DB
(config) # protocol srdf symm id 123 rdf group 2 optimization none description Checkimages
(config) # protocol srdf symm id 123 rdf group 3 optimization sdr-default description VMimages
```

The following shows an example of output:

```
(config) # show protocol srdf symm stats
```

Time	SYMM	RDF group	opt	policy	Reduction	LAN Mbps	WAN Mbps	LAN KB	WAN KB	description
10/2/2010 10:14:49	0123	1		lz-only	68%	222	71.04	222,146	71,040	Oracle1 DB
10/2/2010 10:14:49	0123	2		none	0%	79	79	79,462	79,462	Checkimages

```
10/2/2010 10:14:49 0123 3 sdr-default 94% 299 17.94 299,008 17,943 VMimages
```

Note: Data reduction is highest for RDF Group 3, which is treated with default SDR.

For details of these commands, commands to show the current srdf configuration (**show protocol srdf symm id**), and to show reduction statistics for all or specific SYMMIDs (**show protocol srdf symm stats**), see the *Riverbed Command-Line Interface Reference Manual*.

To fine-tune SRDF optimization on a per-RDF-group basis

1. Check the level of data reduction currently achieved on each RDF group with the **show protocol srdf symm stats** command.
2. For RDF groups achieving low data reduction (for example, less than 20%), change the optimization policy to LZ-only.
3. For RDF groups achieving no data reduction (0%), first check to determine whether the RDF groups contain information that is intentionally encrypted. If so, change the optimization policy to none. If not, Riverbed recommends investigating whether source encryption can be disabled.

Viewing SRDF Reports

In RiOS v7.0 or later, you can report SRDF statistics on a per-RDF-group basis. The following command displays statistics for all RDF groups being optimized:

```
(config) # show stats protocol srdf [interval <interval>] | [start-time <date> end-time <date>]
```

This command shows statistics from a specific Symmetrix machine (indicated by `symm_id`):

```
(config) # show stats protocol srdf symm id <symm_id> [interval <interval>] | [start-time <date> end-time <date>]
```

This command shows statistics from a specific RDF group (indicated by `rdf group`):

```
(config) # show stats protocol srdf symm id <symm_id> rdf-group <rdf group> [interval <interval>] | [start-time <date> end-time <date>]
```

To view these reports, open the Management Console, and choose Reports > Optimization > SRDF.

Best Practices for SAN Replication Using TCP/IP

Many SAN arrays support replication using direct connectivity via TCP/IP. In this case, Steelhead appliances optimize connections that are initiated directly between the SAN arrays participating in the replication. The following table shows a best practice configuration running RiOS v5.5.3 (or later) with TCP/IP connectivity directly from storage array.

Feature	CLI Commands
Enable RiOS SDR-M Note: Optional: When using the Steelhead appliance 7050, select default RiOS SDR for higher data reduction	<code>datastore sdr-policy sdr-m</code>
Set compression level (LZ1)	<code>datastore codec compression level 1</code>
Multicore Balancing	<code>datastore codec multi-core-bal</code>
Enable MX-TCP class covering replication traffic Note: Replace <XXXX> with the port used by the replication application.	<code>qos classification class add class-name "blast" priority realtime min-pct 99.0000000 link-share 100.0000000 upper-limit-pct 100 "root"</code> <code>qos classification rule add class-name "blast" traffic-type optimized destination port <XXXX> rulenum 1</code> <code>qos classification rule add class-name "blast" traffic-type optimized source port <XXXX> rulenum 1</code>
Set WAN TCP buffers	<code>protocol connection wan receive def-buf-size <2*BDP></code> <code>protocol connection wan send def-buf-size <2*BDP></code>
Set LAN TCP buffers	<code>protocol connection lan send buf-size 1048576</code> <code>tcp adv-win-scale -1</code> Note: <code>tcp adv-win-scale -1</code> is for RiOS v5.5.6c or later.
Reset existing connections on start up	<code>in-path kickoff</code> <code>in-path kickoff-resume</code> Note: <code>in-path kickoff-resume</code> is for RiOS v6.0.1a or later.
Never pass-through SYN packets	<code>in-path always-probe enable</code>
Increase encoder buffer sizes	<code>datastore codec multi-codec encoder max-ackqlen 20</code> <code>datastore codec multi-codec encoder global-txn-max 128</code>
SRDF/A optimization Note: Use only with SRDF/A Replication and RiOS v6.0.1 or later.	<code>protocol srdf enable</code>
VMAX DIF header optimization	<code>protocol srdf rule src-ip <x.x.x.x> dst-ip <y.y.y.y> dif enable</code> Replace <x.x.x.x> and <y.y.y.y> with IP address pairs for RE ports. For details, see "Storage Optimization Modules" on page 358 . Note: Use only with EMC VMAX and RiOS v6.0.1 or later.
Restart the optimization service	<code>restart</code>

Best Practices for SAN Replication Using FCIP

FCIP is a transparent FC tunneling protocol over TCP/IP. It can be used across both high and low-speed links, and across long and short distances and latencies. When considering various transports of differing speed and distance, it is important to tune the FCIP transport to ensure expected performance and resiliency, which is directly related to TCP. This section explains some of the design factors you must consider when designing an FCIP SAN.

The following examples deployments described:

- [“Best Practices for a Cisco MDS FCIP Configuration” on page 367](#)
- [“Best Practices for a Brocade 7800 Configuration” on page 371](#)
- [“Best Practices for a Brocade 7500 Configuration” on page 377](#)

Best Practices for a Cisco MDS FCIP Configuration

This section describes the key concepts and recommended settings in the MDS. This section includes the following topics:

- [“FCIP Profiles” on page 367](#)
- [“FCIP Tunnels ” on page 368](#)
- [“Configuring a Cisco MDS FCIP Deployment” on page 368](#)
- [“Best Practices for a RiOS v5.5.3 or Later with Cisco MDS FCIP Configuration” on page 369](#)

FCIP Profiles

An *FCIP profile* defines characteristics of FCIP tunnels that are defined through a particular MDS Gigabit Ethernet interface. Profile characteristics include the:

- IP address of the MDS Gigabit Ethernet interface that is originating the tunnel.
- TCP port number.
- bandwidth and latency characteristics of the WAN link.
- advanced settings that are typically left to their default values.

The MDS enables you to define up to three FCIP profiles per physical MDS Gigabit Ethernet interface. Because a tunnel can be created for each profile, a Cisco MDS switch with two physical Gigabit Ethernet ports can have up to six profiles. Most configurations have only one profile per Gigabit Ethernet interface. Riverbed recommends maximizing the number of profiles configured for each GigE port to increase the total number of TCP connections.

In the profile setting, the default maximum and minimum bandwidth settings per FCIP profile are 1000 Mbps and 500 Mbps, respectively. You can achieve better performance for unoptimized and optimized traffic using 1000 Mbps and 800 Mbps. This is the rate of the LAN-side TCP entering the Steelhead appliance, so that setting it aggressively high does not have any downside, because the Steelhead appliance terminates TCP locally on the LAN-side and the MDS can slow down if it tries to go too fast by advertising a smaller TCP window.

Similarly, leave the round-trip setting at its default (1000 msec in the Management Console, 1 ms in the CLI), because the *network* in this context is effectively the LAN connection between the MDS and the Steelhead appliance.

If you are doing *unoptimized runs*, configure the bandwidth and latency settings in the MDS to reflect the actual network conditions of the WAN link. These settings improve performance in terms of enabling the MDS to fill-the-pipe with unoptimized runs in the presence of latency.

FCIP Tunnels

An FCIP *tunnel* configuration is attached to a profile and defines the IP address and TCP port number of a far-side MDS to which an FCIP connection is established. You can keep the tunnel configuration default settings, with the following key exceptions:

- In the Advanced tab of the MDS GUI:
 - Turn on the Write Accelerator option. Always use this option when testing with Steelhead appliances in the presence of latency. This is an optimization in the MDS (and similar features exist in other FCIP/iFCP products) to reduce round trips.
 - Set the FCIP configuration for each tunnel to Passive on one of the MDS switches. By default, when first establishing FCIP connectivity, each MDS normally tries to constantly initiate new connections in both directions, and it is difficult to determine which side ends up with the well-known destination port (for example, 3225). This behavior can make it difficult to interpret Steelhead appliance reports. When you set one side to Passive, the non-passive side always initiates connections, hence the behavior is deterministic.

FCIP settings allow you to specify the number of TCP connections associated with each FCIP tunnel. By default, this setting is 2: one for Control traffic, and one for the Data traffic. Do not change the default value. The single-TCP mode only exists to maintain compatibility with older FCIP implementations. Separating the Control and Data traffic has performance implications because FC is highly jitter-sensitive.

Finally, you can set whether the MDS compresses the FCIP data within the FCIP tunnel configuration. You must disable it when the Steelhead appliance is optimizing. On the MDS the default setting is off. The best practices of common SAN replication vendors (for example, EMC) recommend turning on this setting when there are no WAN optimization controller (WOC) systems present. However, when adding Steelhead appliances to an existing environment, it should be disabled.

Configuring a Cisco MDS FCIP Deployment

The following example shows a Cisco MDS FCIP gateway configuration. Cisco-style configurations, typically does not show the default values (for example, compression is off by default, and is not present in this configuration dump). Also, this configuration does not show any non-FCIP elements (such as the FC ports that connect to the SAN storage array and VSANs). This example shows a standard and basic topology that includes an MDS FCIP gateway at each end of a WAN link, MDS1, and MDS2.

To configure a standard and basic topology that includes an MDS FCIP gateway

1. Configure MDS1.

```
fcip profile 1
  ip address 10.12.254.15
  tcp max-bandwidth-mbps 1000 min-available-bandwidth-mbps 800 round-trip-time-ms 1
fcip profile 2
  ip address 10.12.254.145
  tcp max-bandwidth-mbps 1000 min-available-bandwidth-mbps 800 round-trip-time-ms 1
interface fcip1
  use-profile 1
  peer-info ipaddr 10.12.254.45
  write-accelerator
  no shutdown
interface fcip2
  use-profile 2
```

```

peer-info ipaddr 10.12.254.245
write-accelerator
no shutdown
ip route 10.12.254.32 255.255.255.224 10.12.254.30
ip route 10.12.254.224 255.255.255.224 10.12.254.130
interface GigabitEthernet1/1
ip address 10.12.254.15 255.255.255.224
switchport description LAN side of mv-emcsh1
no shutdown
interface GigabitEthernet1/2
ip address 10.12.254.145 255.255.255.224
switchport description LAN side of mv-emcsh1
no shutdown

```

2. Configure MDS2.

```

fcip profile 1
ip address 10.12.254.45
tcp max-bandwidth-mbps 1000 min-available-bandwidth-mbps 800 round-trip-time-ms 1
fcip profile 2
ip address 10.12.254.245
tcp max-bandwidth-mbps 1000 min-available-bandwidth-mbps 800 round-trip-time-ms 1
interface fcip1
use-profile 1
passive-mode
peer-info ipaddr 10.12.254.15
write-accelerator
no shutdown
interface fcip2
use-profile 2
passive-mode
peer-info ipaddr 10.12.254.145
write-accelerator
no shutdown
ip route 10.12.254.0 255.255.255.224 10.12.254.60
ip route 10.12.254.128 255.255.255.224 10.12.254.230
interface GigabitEthernet1/1
ip address 10.12.254.45 255.255.255.224
switchport description LAN side of mv-emcsh2
no shutdown
interface GigabitEthernet1/2
ip address 10.12.254.245 255.255.255.224
switchport description LAN side of mv-emcsh2
no shutdown

```

Best Practices for a RiOS v5.5.3 or Later with Cisco MDS FCIP Configuration

Riverbed recommends the following best practices regarding a Cisco MDS FCIP configuration:

- Enable the RiOS v5.5 or later multicore balancing feature due to the small number of data connections.
- Use an in-path rule to specify the neural-mode as *never* for FCIP traffic.
- Set the *always-probe* port to 3225 to ensure that MDS aggressive SYN-sending behavior does not result in unwanted pass-through connections.

The following table summarizes the CLI commands RiOS v5.5.3 or later with Cisco MDS FCIP.

Feature	CLI Commands
Enable RiOS SDR-M Optional: When using the Steelhead appliance 7050, select default RiOS SDR for higher data reduction.	datastore sdr-policy sdr-m
Set compression level (LZ1)	datastore codec compression level 1
Multicore Balancing	datastore codec multi-core-bal
Turn Off Nagle	in-path rule auto-discover srcaddr all-ip dstaddr all-ip dstport "3225" preoptimization "none" optimization "normal" latency-opt "normal" vlan -1 neural-mode "never" wan-visibility "correct" description "" rulenum start
MX-TCP class covering FCIP traffic	qos classification class add class-name "blast" priority realtime min-pct 99.0000000 link-share 100.0000000 upper-limit-pct 100.0000000 queue-type mxtcp queue-length 100 parent "root" qos classification rule add class-name "blast" traffic-type optimized destination port 3225 rulenum 1 qos classification rule add class-name "blast" traffic-type optimized source port 3225 rulenum 1
Set WAN TCP buffers	protocol connection wan receive def-buf-size <2*BDP> protocol connection wan send def-buf-size <2*BDP>
Set LAN TCP buffers	protocol connection lan send buf-size 1048576 tcp adv-win-scale -1 Note: tcp adv-win-scale -1 is for RiOS v5.5.6c or later.
Reset existing connections on startup	in-path kickoff in-path kickoff-resume Note: in-path kickoff-resume is for RiOS v6.0.1a or later.
Never pass-through SYN packets	in-path always-probe enable
Change always-probe port to FCIP	in-path always-probe port 3225
Increase encoder buffer sizes	datastore codec multi-codec encoder max-ackqlen 20 datastore codec multi-codec encoder global-txn-max 128
FCIP optimization	protocol fcip enable Note: Use only with RiOS v6.0.1 or later.
DIF header optimization	protocol fcip rule src-ip <x.x.x.x> dst-ip <y.y.y.y> dif enable Replace <x.x.x.x> and <y.y.y.y> with IP address pairs for MDS Gigabit Ethernet ports. For details, see "Storage Optimization Modules" on page 358. Note: Use only with EMC VMAX and RiOS v6.0.1 or later.
Restart the optimization service	restart

If you increase the number of FCIP profiles, you must also create separate in-path rules to disable Nagle for other TCP ports (for example, 3226 and 3227).

Similarly, if you decide to set QoS rules to focus on port 3225 to drive traffic into a particular class, you must create rules for both ports 3226 and 3227. Riverbed does not recommend a multiprofile-per-GigE-port configuration.

Best Practices for a Brocade 7800 Configuration

This section provides best practice configuration requirements to deploy Steelhead appliances for use in optimizing Brocade 7800 FCIP solutions. This section includes the following topics:

- [“Configuring the Steelhead Appliance for Brocade 7800” on page 371](#)
- [“Configuring FCIP Tunnels on Brocade 7800” on page 373](#)

Successful deployment of the combined solution requires use of qualified versions of hardware and software from both Riverbed and Brocade. For current information about Brocade hardware and software qualified for use with Steelhead appliances, see the Knowledge Base article *Deploying Steelhead Appliances with Brocade*, at <https://supportkb.riverbed.com/support/index?page=content&id=S13362>.

Configuring the Steelhead Appliance for Brocade 7800

Use the following settings on the Steelhead appliance:

- Enable the FCIP optimization module.
- Enable a FCIP DIF rule (if necessary) when optimizing EMC Symmetrix VMAX FC-based (RF port) SRDF traffic.
- Disable neural framing for TCP port 3225.
- Increase TCP buffer space allocated for overhead.
- Decrease TCP timer to 45 seconds.
- Use the **always probe enable** CLI command for TCP port 3225.
- Enable multiple core processing.
- Enable SDR-M (except for Steelhead appliance 7050/701).
- Enable MXTCP.
- Configure the WAN buffers to 2*BDP (bandwidth-delay product) and set LAN buffers to default.
- Enable LZ1 compression (except for Steelhead appliance 7050/701).
- Set in-path kick off.
- Enable fail-to-block (optional).

Keep the following caveats and additional notes in mind.

Caveats and Additional Notes	CLI Command
The Brocade 7800 extension switch uses TCP port 3225 for its FCIP flows. Enable FCIP optimization for port 3225. The default ports for FCIP optimization are 3225, 3226, 3227, and 3228.	protocol fcip enable
Enable FCIP DIF header isolation (if necessary) when optimizing EMC Symmetrix VMAX FC-based (RF port) SRDF traffic. The example CLI command isolates DIF headers after every 512 bytes of storage traffic, which is typical for Open Systems.	protocol fcip rule src-ip 0.0.0.0 dst-ip 0.0.0.0 dif enable dif-blocksize 512 enc-header enable

Caveats and Additional Notes	CLI Command
In testing, disabling neural framing increases throughput for FCIP traffic. If you disable neural framing, you allow a faster response of SCSI layer acknowledgements from the target device.	in-path rule auto-discover srcaddr all-ip dstaddr all-ip dstport dstport 3225 neural-mode never
Due to the way that the Brocade 7800 extension switch sends traffic, if you increase the TCP buffer space reserved for overhead, you have a more stable throughput. The CLI command is hidden and does not auto-complete.	tcp adv-win-scale -1
The Brocade 7800 extension switch tends to use the same source TCP port to setup a connection. In cases where error recovery is required, you must align the timer on the Steelhead appliance that holds on to optimized connection with the timer of the Brocade 7800 extension switch. By default, the Steelhead appliance holds on to idle connections for 15 minutes. Adjust the timer to less than 1 minute. The example CLI commands shows how to adjust the TCP time-out value to 45 seconds.	tcp max-time-out 45 tcp max-time-out mode enable
The Brocade 7800 extension switch aggressively sends SYN packets to quickly establish TCP connections. Because of this behavior, the Steelhead appliance might pass through the connection. To ensure that the Steelhead appliance attempts to optimize every SYN from the Brocade 7800 extension switch, use the following CLI commands to force probing on TCP port 3225.	in-path always-probe enable in-path always-probe port 3225
By default, a Steelhead appliance uses only the resources of a single CPU core to process a single TCP stream. Because FCIP traffic is carried among a small number of TCP streams, enable the RiOS multiple core feature for more efficient use of additional CPU cores on the Steelhead appliance. When you use this command, you further increases end-to-end throughput.	datastore codec multi-core-bal
Riverbed recommends that you enable SDR-M to achieve the highest throughput in FCIP environments, except on the 7050/701. The 7050/701 achieves high throughput with standard SDR due to its SSD-based data store.	datastore sdr-policy sdr-m
Configure MXTCP for Brocade FCIP traffic on TCP port 3225 for best throughput.	qos classification class add class-name "blast" priority realtime min-pct 99.0000000 link-share 100.0000000 upper-limit-pct 100.0000000 queue-type mxtcp queue-length 100 parent "root" qos classification rule add class-name "blast" traffic-type optimized destination port 3225 rulenum 1
Set the WAN buffer to 2*BDP (bandwidth-delay product) using the following formula. Do not change the default settings on LAN buffers, because by default, LAN buffers auto-tune to the appropriate size to provide the best throughput. RTT is the round-trip time (ping time) in ms. $\frac{[\text{Bandwidth bps}] \times [\text{RTT ms}] \times 2}{8 \times 1000}$ The following example shows how to calculate the WAN buffer setting for a 155 Mbps link with 100 ms of RTT: $155,000,000 \times 100 \times 2 / (8 \times 1000) = 3875000 \text{ bytes}$	protocol connection wan receive def-buf-size 3875000 protocol connection wan send def-buf-size 3875000
Use LZ1 compression to achieve the best throughput on all models below the 7050/701. Because compression on the 7050/701 is done in hardware, high throughput is achieved with a higher default compression level.	datastore codec compression level 1

Caveats and Additional Notes	CLI Command
Enable in-path kickoff so pass through FCIP connections are reset and re-optimized when service on a Steelhead appliance is restarted.	in-path kickoff in-path kickoff-resume
<p>You can choose to not send FCIP traffic when the Steelhead appliance is not available to optimize the FCIP flows. Your network might not have the bandwidth to provide the throughput that is necessary for replication traffic, which can cause a replication traffic error.</p> <p>You can configure the Steelhead appliance to block all traffic in cases in which the service is stopped on the Steelhead appliance, or if the Steelhead appliance loses power. In the following example the command is applied to the inpath0_0 interface.</p>	no interface inpath0_0 fail-to-bypass enable

Configuring FCIP Tunnels on Brocade 7800

The Brocade 7800 extension switch introduces the concept of trunking. Trunking allows load-balancing and failover capabilities through the use of FCIP circuits. FCIP circuits are combined to form one FCIP tunnel. When you use trunking with the Steelhead appliance, you have a better throughput because the load is spread among multiple TCP connections and Gigabit Ethernet interfaces.

FCIP circuits have the following attributes:

- An FCIP circuit is a connection between a pair of IP addresses.
- You can assign each FCIP circuit a cost.
- Traffic load balances only between lowest cost circuits.
- If a lowest cost FCIP circuit fails, than higher cost FCIP circuits are used.
- Each FCIP circuit can have maximum commit rate of 1 Gbps.
- You can combine up to four FCIP circuits into one FCIP tunnel.
- You can define all FCIP circuits on one Gigabit Ethernet interface or spread among up to four Gigabit Ethernet interfaces.
- IP addresses of Gigabit Ethernet interfaces must be in different subnets.
- Each FCIP circuit can have four TCP connections for different priorities: high, medium, low and class F traffic.
- By default, if you do not configure QoS on the Brocade 7800 extension switch, all user traffic falls in the medium priority TCP connection, and switch control traffic uses the class F TCP connection.

Use the following settings on the Brocade 7800 extension switch for use with the Steelhead appliance:

- Configure FCIP trunking with four circuits, spread amongst available Gigabit Ethernet interfaces.
- Disable FCIP compression.
- Enable FCIP Fast Write.
- Set the FCIP commit rate to 900 Mbps total per Gigabit Ethernet interface.

The following example shows that four FCIP circuits are defined between two Gigabit Ethernet interfaces: Gigabit Ethernet0 and Gigabit Ethernet1. The Brocade 7800 extension switch has six Gigabit Ethernet interfaces.

To configure FCIP tunnels on Brocade 7800

1. On the Brocade 7800 extension switch, connect to the CLI.

- Assign an IP address to the Gigabit Ethernet 0 (ge0) interface. Because two FCIP circuits are defined on this interface, you must configure two IP addresses. Both IP addresses must be on the same subnet. The IP addresses on ge1 must be on a different subnet than ge0.

```
mv-broc5 >>portcfg ipif ge0 create 10.12.254.3 255.255.255.224 1500
mv-broc5 >>portcfg ipif ge0 create 10.12.254.6 255.255.255.224 1500
```

Where the following is true:

- Interface IP being applied to = ge0
- IP address assigned to interface = 10.12.254.3 / 10.12.254.6
- Network Mask = 255.255.255.224
- MTU size = 1500

- Repeat for Gigabit Ethernet 1 (ge1).

```
mv-broc5 >>portcfg ipif ge1 create 10.12.254.137 255.255.255.224 1500
mv-broc5 >>portcfg ipif ge1 create 10.12.254.133 255.255.255.224 1500
```

Where the following is true:

- Interface IP being applied to = ge1
- IP address assigned to interface = 10.12.254.137 / 10.12.254.133
- Network Mask = 255.255.255.224
- MTU size = 1500

- To verify the IP configuration on ge0, enter the following command:

```
mv-broc5> portshow ipif ge0
Port: ge0
Interface IPv4 Address      NetMask      Effective MTU  Flags
-----
0          10.12.254.3    255.255.255.224 1500          U R M
1          10.12.254.6    255.255.255.224 1500          U R M
```

- Create a static route for an FCIP tunnel if the peer IP address of the remote Brocade 7800 extension switch is on a different subnet by entering the following command:

```
mv-broc5> portcfg iproute ge0 create 10.12.254.32 255.255.255.224 10.12.254.30
Operation Succeeded
```

where the following is true:

- Interface = ge0
- Destination Network = 10.12.254.32
- Destination IP Mask = 255.255.255.224
- Next Hop IP = 10.12.254.30

Use the following command to verify the static route configured on a Gigabit Ethernet interface:

```
mv-broc5> portshow iproute ge0

Port: ge0
IP Address      Mask      Gateway      Metric  Flags
-----
10.12.254.0     255.255.255.224 *          0       U C
10.12.254.6     255.255.255.255 *          0       U C
10.12.254.30    255.255.255.255 *          0       U H L
10.12.254.32    255.255.255.224 10.12.254.30 0       U G S
```


6. Repeat this step for ge1.
7. Enable virtual ports for the Gigabit Ethernet interface.

When a FCIP tunnel is defined, each FCIP tunnel is associated with virtual port. You must specify the unique virtual port the FCIP tunnel is associated with.

```
mv-broc6 >>portcfgpersistenable 16
```

Where the following is true:

- Virtual ports range from 16 to 23

8. Create the FCIP tunnel with four circuits by entering the following commands.

The following example shows four circuits created across two Gigabit Ethernet interfaces. For each Gigabit Ethernet interface the total commit rate of the circuits is equal to a total of 900 Mbps. Each Gigabit Ethernet interface has two circuits, so for each circuit the commit rate is set to 450 Mbps. The first two commands create circuits with the IP addresses of ge0, and the last two commands create circuits with the IP addresses of ge1.

```
mv-broc5 >> portcfg fcip tunnel 16 create 10.12.254.35 10.12.254.3 450000 -f
mv-broc5 >> portcfg fcip circuit 16 create 1 10.12.254.36 10.12.254.6 450000
mv-broc5 >> portcfg fcip circuit 16 create 2 10.12.254.233 10.12.254.133 450000
mv-broc5 >> portcfg fcip circuit 16 create 3 10.12.254.237 10.12.254.137 450000
```

Where the following is true:

- Virtual port number = 16
- Destination IP = remote peer IP address
- Source IP = local IP address
- comm_rate in kilobits = 450000
- -f (enables fastwrite)

To modify a parameter on any FCIP tunnel that has been created, use the **modify** option of the **portcfg** command.

The following table has commands to show the state of the FCIP tunnel and to diagnose issues with Brocade 7800.

Description	Command
Allows a ping to be sourced from a specified interface. This is useful for verifying connectivity between two Brocade 7800 extension switches.	<pre>mv-broc5> portcmd --ping ge0 -s 10.12.254.3 -d 10.12.254.35 Pinging 10.12.254.35 from ip interface 10.12.254.3 on 0/ge0 with 64 bytes of data Reply from 10.12.254.35: bytes=64 rtt=0ms ttl=97 Reply from 10.12.254.35: bytes=64 rtt=0ms ttl=97 Ping Statistics for 10.12.254.35: Packets: Sent = 4, Received = 4, Loss = 0 (0 percent loss) Min RTT = 0ms, Max RTT = 0ms Average = 0ms</pre>
Shows the state of a Gigabit Ethernet interface.	<pre>mv-broc5> portshow ge0 Eth Mac Address: 00.05.1e.a5.55.da Port State: 1 Online Port Phys: 6 In_Sync Port Flags: 0x4003 PRESENT ACTIVE LED Port Speed: 1G</pre>

Description	Command
Shows the mode of a specific Gigabit Ethernet interface. In the following example the Gigabit Ethernet is set for FCIP.	<pre>mv-broc5> portshow mode ge0 Port: ge0 ----- Mode: FCIP</pre>
Shows all the switches in the fabric. This is useful when verifying FCIP connectivity. The following example shows output for two switches.	<pre>mv-broc5> fabricshow Switch ID Worldwide Name Enet IP Addr FC IP Addr Name ----- 3: fffc03 10:00:00:05:1e:a5:55:d6 10.2.224.35 0.0.0.0 >"mv-broc5" 4: fffc04 10:00:00:05:1e:c3:cb:8f 10.2.224.36 0.0.0.0 "mv-broc6" The Fabric has 2 switches</pre>
Shows the state of the switch. This is useful for looking at all the state of all interfaces on the Brocade 7800 extension switch.	<pre>mv-broc5> switchshow switchName: mv-broc5 switchType: 83.3 switchState: Online switchMode: Native switchRole: Principal switchDomain: 3 switchId: fffc03 switchWwn: 10:00:00:05:1e:a5:55:d6 zoning: ON (fcip_zoneconfig) switchBeacon: OFF FC Router: OFF FC Router BB Fabric ID: 1 Address Mode: 0 Index Port Address Media Speed State Proto ===== 0 0 030000 id N4 Online FC F-Port 50:00:09:72:c0:05:ad:5c 1 1 030100 id N4 Online FC F-Port 50:00:09:72:c0:05:ad:58 2 2 030200 id N8 No_Light FC Disabled (Persistent) 3 3 030300 id N8 No_Light FC 4 4 030400 id N8 No_Light FC 5 5 030500 id N8 No_Light FC 6 6 030600 id N8 No_Light FC 7 7 030700 id N8 No_Light FC 8 8 030800 -- N8 No_Module FC 9 9 030900 -- N8 No_Module FC 10 10 030a00 -- N8 No_Module FC 11 11 030b00 -- N8 No_Module FC 12 12 030c00 -- N8 No_Module FC 13 13 030d00 -- N8 No_Module FC 14 14 030e00 -- N8 No_Module FC 15 15 030f00 -- N8 No_Module FC 16 16 031000 -- -- Online VE VE-Port 10:00:00:05:1e:c3:cb:8f "mv-broc6" (downstream) 17 17 031100 -- -- Offline VE 18 18 031200 -- -- Offline VE 19 19 031300 -- -- Offline VE 20 20 031400 -- -- Offline VE 21 21 031500 -- -- Offline VE 22 22 031600 -- -- Offline VE 23 23 031700 -- -- Offline VE ge0 cu 1G Online FCIP Copper ge1 cu 1G Online FCIP Copper ge2 -- 1G No_Module FCIP ge3 -- 1G No_Module FCIP ge4 -- 1G No_Module FCIP ge5 -- 1G No_Module FCIP</pre>

Best Practices for a Brocade 7500 Configuration

This section provides best practice configuration requirements to deploy Steelhead appliances for use in optimizing Brocade 7500 FCIP solutions. This section includes the following topics:

- [“Configuring the Steelhead Appliance for Brocade 7500” on page 377](#)
- [“Configuring FCIP Tunnels on Brocade 7500” on page 377](#)
- [“Best Practices for RiOS v5.5.6c or Later with Brocade 7500” on page 379](#)

Configuring the Steelhead Appliance for Brocade 7500

Successful deployment of the combined solution requires use of qualified versions of hardware and software from both Riverbed and Brocade. For current information about Brocade hardware and software qualified for use with Steelhead appliances, see the Knowledge Base article *Deploying Steelhead Appliances with Brocade*, at <https://supportkb.riverbed.com/support/index?page=content&id=S13362>.

Configuring FCIP Tunnels on Brocade 7500

Use the following settings for the Brocade 7500 extension switch:

- Compression disabled
- FCIP Fastwrite enabled
- FCIP bandwidth set to 900 Mbps
- One FCIP tunnel on one Gigabit Ethernet Interface
- Byte streaming mode enabled (required)

If you are installing Steelhead appliances into an existing FCIP SAN extension configuration where previously there were no WOC systems present, some of these settings might be different must be changed.

To configure FCIP tunnels on Brocade 7500

1. On the Brocade 7500 extension switch, connect to the CLI.
2. Assign an IP address to a Gigabit Ethernet interface by entering the following command (the Brocade 7500 extension switch has Gigabit Ethernet interfaces ge0 and ge1):

```
portcfg ipif ge0 create 11.1.1.2 255.255.255.0 1500
```

Where the following is true:

- Interface IP being applied to = ge0
- IP address assigned to interface = 11.1.1.2
- Network Mask = 255.255.255.0
- MTU size = 1500

3. To verify the IP configuration on ge0, enter the following command:

```
portshow ipif ge0
```

4. Create a static route for an FCIP tunnel if the peer IP address of the remote Brocade 7500 extension switch is on a different subnet by entering the following command:

```
portcfg iproute ge0 create 12.1.1.0 255.255.255.0 11.1.1.1
```

Where the following is true:

- Interface = ge0
- Destination Network = 12.1.1.0
- Destination IP Mask = 255.255.255.0
- Next-hop IP = 11.1.1.1

5. To verify the static route configured on ge0, enter the following command:

```
portshow iproute ge0
```

6. Enable virtual port for the Gigabit Ethernet interface.

When a FCIP tunnel is defined, each FCIP tunnel is associated to a virtual port. Each Gigabit Ethernet interface can have up to eight FCIP tunnels defined: tunnel ID 0-7. On Gigabit Ethernet 0, tunnels 0 through 7 are tied to virtual ports 16 to 23. On Gigabit Ethernet 1, tunnels 0 through 7 are tied to virtual ports 24 thru 31. Enable the associated virtual port with the FCIP tunnel ID defined. To create tunnel 0 on physical port Gigabit Ethernet 0, connect to the CLI and enter the following command:

```
portcfgpersistentenable 16
```

Where the following is true:

- Virtual ports 16-23 correspond to ge0 tunnels 0-7
- Virtual ports 24-31 correspond to ge1 tunnels 0-7

The Brocade 7500 extension switch does not allow multiple equal-cost FCIP tunnels in the same zone when using FCIP Fastwrite. Because FCIP Fastwrite is essential for the end-to-end WAN throughput when using Steelhead appliance optimization, the best practice configuration uses only a single FCIP tunnel on a single Brocade Gigabit Ethernet port.

7. Create the FCIP tunnel on ge0 by entering the following command:

```
portcfg fciptunnel ge0 create 0 11.1.1.4 11.1.1.2 900000 -f -bstr
```

Where the following is true:

- Interface = ge0
- Tunnel ID = 0
- Destination IP = 11.1.1.4
- Source IP = 11.1.1.2
- Committed Rate (comm._rate) in kilobits = 900 Mbps
- -f (enables FCIP Fastwrite)
- -bstr (enables byte streaming mode)

By default compression is disabled and cannot be enabled when byte streaming mode is configured.

To modify a parameter on any FCIP tunnel that has been created, use the **modify** option of the **portcfg** command. In the following example the committed rate on the FCIP tunnel is modified to 800 Mbps.

```
portcfg fciptunnel ge0 modify 0 11.1.1.4 800000
```

Best Practices for RiOS v5.5.6c or Later with Brocade 7500

This section provides the best practices regarding a Brocade 7500 configuration:

- Enable the RiOS Multicore Balancing feature due to the small number of data connections.
- Turn off Nagle on TCP port 3226 to significantly increase FCIP performance because of the claimed latency/jitter-sensitivity of FC/FCIP.
- Set the *always-probe* port to 3226. The Brocade 7500 extension switch has an aggressive SYN-sending behavior that can sometimes cause pass-through connections.
- The Brocade 7500 extension switch does not support closing TCP connections with FINs. Instead, it only uses RSTs.
- The Brocade 7500 extension switch uses two different TCP connections for one FCIP tunnel. TCP port 3225 is used for control and port 3226 is used for data. Be advised that port 3225 is passed through by the Steelhead appliance because the control connection typically does not have data to reduce.
- Increase the TCP buffer space reserved for overhead. Due to the way that the Brocade 7500 extension switch sends traffic, increasing the TCP buffer space reserved for overhead supplies a more stable throughput.
- Because the Brocade 7500 extension switch tends to use the same source TCP port to set up a connection, in cases where error recovery is required, the timer on the Steelhead appliance that holds on to optimized connections needs to be better aligned with the Brocade 7500 extension switch. Steelhead appliances by default hold on to idle connections for 15 minutes. Adjust the time to be less than one minute.

The following table summarizes the CLI commands for RiOS v5.5.6c or later with Brocade 7500.

Feature	CLI Commands
Enable RiOS SDR-M Optional: When using the Steelhead appliance 7050, select default RiOS SDR for higher data reduction.	<code>datastore sdr-policy sdr-m</code>
Set compression level (LZ1)	<code>datastore codec compression level 1</code>
Multicore Balancing	<code>datastore codec multi-core-bal</code>
Turn Off Nagle	<code>in-path rule auto-discover srcaddr all-ip dstaddr all-ip dstport "3226" preoptimization "none" optimization "normal" latency-opt "normal" vlan -1 neural-mode "never" wan-visibility "correct" description "" rulenum start</code>
MX-TCP class for FCIP traffic	<code>qos classification class add class-name "blast" priority realtime min-pct 99.0000000 link-share 100.0000000 upper-limit-pct 100.0000000 queue-type mxtcp queue-length 100 parent "root" qos classification rule add class-name "blast" traffic-type optimized destination port 3226 rulenum 1 qos classification rule add class-name "blast" traffic-type optimized source port 3226 rulenum 1</code>
Set WAN TCP buffers	<code>protocol connection wan receive def-buf-size <2*BDP> protocol connection wan send def-buf-size <2*BDP></code>

Feature	CLI Commands
Set LAN TCP buffers	protocol connection lan send buf-size 1048576 tcp adv-win-scale -1 Note: tcp adv-win-scale -1 is for RiOS v5.5.6c or later.
Use RST to terminate connections	sport splice-policy outer-rst-port port 3226
Pass-through control traffic	in-path rule pass-through srcaddr all-ip dstaddr all-ip dstport "3225" vlan -1 description "" rulenum start
Reduce TCP timeout value	tcp max-time-out 45 tcp max-time-out mode enable
Reset existing connections on startup	in-path kickoff in-path kickoff-resume (RiOS v6.0.1a+ only)
Never pass-through SYN packets	in-path always-probe enable
Change always-probe port to FCIP	in-path always-probe port 3226
Increase encoder buffer sizes	datastore codec multi-codec encoder max-ackqlen 20 datastore codec multi-codec encoder global-txn-max 128
FCIP optimization	protocol fcip enable Note: Use only with RiOS v6.0.1 or later.
DIF header optimization	protocol fcip rule src-ip <x.x.x.x> dst-ip <y.y.y.y> dif enable Replace <x.x.x.x> and <y.y.y.y> with IP address pairs for MDS Gigabit Ethernet ports. For details, see "Storage Optimization Modules" on page 358 . Note: Use only with EMC VMAX and RiOS v6.0.1 or later.
Restart the optimization service	restart


CHAPTER 18 Proxy File Services Deployments

This chapter describes proxy file services (PFS) and the basic steps for configuring PFS. This chapter includes the following sections:

- [“Overview of Proxy File Services” on page 381](#)
- [“Upgrading v2.x PFS Shares” on page 383](#)
- [“Domain and Local Workgroup Settings” on page 384](#)
- [“PFS Share Operating Modes” on page 386](#)
- [“Configuring PFS” on page 387](#)

Important: Do not configure both RSP and PFS on the same Steelhead appliance. Riverbed does not support this configuration because PFS has no disk boundaries and can overwrite the space allocated to RSP.

Note: PFS is supported on Steelhead appliance models 150, 250, 550, 1050, and 2050.

 VSH does not support PFS.

Overview of Proxy File Services

This section describes Proxy File Services (PFS) and how it works. This section includes the following topics:

- [“When to Use PFS” on page 382](#)
- [“PFS Terms” on page 383](#)

PFS is an integrated virtual file server that allows you to store copies of files on the Steelhead appliance with Windows file access, creating several options for transmitting data between remote offices and centralized locations with improved performance. Data is configured into file shares that are periodically synchronized transparently in the background, over the optimized connection of the Steelhead appliance. PFS leverages the integrated disk capacity of the Steelhead appliance to store file-based data in a format that allows it to be retrieved by NAS clients.

When to Use PFS

Before you configure PFS, evaluate whether it is suitable for your network needs. The advantages of using PFS are:

- **LAN access to data residing across the WAN** - File access performance is improved between central and remote locations. PFS creates an integrated file server, enabling clients to access data directly from the proxy filer on the LAN instead of the WAN. Transparently in the background, data on the proxy filer is synchronized with data from the origin file server over the WAN.
- **Continuous access to files in the event of WAN disruption** - PFS provides support for disconnected operations. In the event of a network disruption that prevents access over the WAN to the origin server, files can still be accessed on the local Steelhead appliance.
- **Simple branch infrastructure and backup architectures** - PFS consolidates file servers and local tape backup from the branch into the data center. PFS enables a reduction in number and size of backup windows running in complex backup architectures.
- **Automatic content distribution** - PFS provides a means for automatically distributing new and changed content throughout a network.

If any of these advantages can benefit your environment, then enabling PFS in the Steelhead appliance is appropriate.

However, PFS requires pre-identification of files and is not appropriate in environments in which there is concurrent read-write access to data from multiple sites:

- **Pre-identification of PFS files** - PFS requires that files accessed over the WAN are identified in advance. If the data set accessed by the remote users is larger than the specified capacity of your Steelhead appliance model or if it cannot be identified in advance, end-users must access the origin server directly through the Steelhead appliance without PFS. (This configuration is also referred to as *Global mode*.)
- **Concurrent read-write data access from multiple sites** - In a network environment where users from multiple branch offices update a common set of centralized files and records over the WAN, the Steelhead appliance without PFS is the most appropriate solution because file locking is directed between the client and the server. The Steelhead appliance always consults the origin server in response to a client request; it never provides a proxy response or data from its RiOS data store without consulting the origin server.

PFS Terms

The following terms are used to describe PFS processes and devices.

PFS Term	Description
Proxy file server	A virtual file server that resides on the Steelhead appliance and provides Windows file access (with ACLs) capability at a branch office on the LAN. The proxy file server is populated over an optimized WAN connection with data from the origin server.
Origin file server	A server located in the data center that hosts the origin data volumes.
Domain mode	A PFS configuration in which the Steelhead appliance joins a Windows domain (typically your company domain) as a member.
Domain controller (DC)	The host that provides user login service in the domain. (Typically, with Windows 2000 Active Directory Service domains, given a domain name, the system automatically retrieves the DC name.)
Local workgroup mode	A PFS configuration in which you define a workgroup and add individual users who have access to the PFS shares on the Steelhead appliance.
Share	The data volume exported from the origin server to the remote Steelhead appliance. Important: The PFS share and the origin-server share name cannot contain Unicode characters. The Management Console does not support Unicode characters.
Local name	The name that you assign to a share on the Steelhead appliance. This is the name by which users identify and map a share. Important: The PFS share and the origin-server share name cannot contain Unicode characters. The Management Console does not support Unicode characters.
Remote path	The path to the data on the origin server or the UNC path of a share you want to make available to PFS.
Share synchronization	The process by which data on the proxy file server is synchronized with the origin server. Synchronization runs periodically in the background, based on your configuration. You can configure the Steelhead appliance to refresh the data automatically at an interval you specify or manually at any time. The levels of synchronization are as follows: <ul style="list-style-type: none"> • Incremental Synchronization - In incremental synchronization, only new and changed data is sent between the proxy file server and the origin file server. • Full Synchronization - In full synchronization, a full directory comparison is performed. The last full synchronization is sent between the proxy file server and the origin file server.

Upgrading v2.x PFS Shares

By default, when you configure PFS shares with Steelhead appliance software v3.x or later, you create v3.x PFS shares. PFS shares configured with Steelhead appliance software v2.x are v2.x shares. V2.x shares are not upgraded when you upgrade Steelhead appliance software.

If you have shares created with v2.x software, Riverbed recommends that you upgrade them to v3.x shares in the Management Console. If you upgrade any v2.x shares, you must upgrade all of them. After you have upgraded shares to v3.x, you can only create v3.x shares.

If you do not upgrade your v2.x shares:

- Do not create v3.x shares.
- Install and start the RCU on the origin server or on a separate Windows host with write-access to the data PFS uses. The account that starts the RCU must have write permissions to the folder on the origin file server that contains the data PFS uses. You can download the RCU from the Riverbed Support site at <https://support.riverbed.com>. For details, see the *Riverbed Copy Utility Reference Manual*.

In Steelhead appliance software v3.x or later, you do not need to install the RCU service on the server for synchronization purposes. All RCU functionality has been moved to the Steelhead appliance.

- Configure domain settings, not workgroup settings, as described in [“Domain and Local Workgroup Settings” on page 384](#). Domain mode supports v2.x PFS shares but Workgroup mode does not.

For details, see the *Steelhead Appliance Management Console User’s Guide*.

Note: Starting from RiOS v7.0.0, PFS shares are synced using a CIFS client on the Steelhead appliance. This mode of syncing does not support PFS v2 shares. RiOS upgrades any PFS v2 shares to v3 when you upgrade the Steelhead appliance image to RiOS v7.0.0 or later. The option to create v2 shares is provided for legacy purposes and might not always be available.

Domain and Local Workgroup Settings

When you configure your PFS Steelhead appliance, set either domain or local workgroup settings. This section includes the following topics:

- [“Domain Mode” on page 384](#)
- [“Local Workgroup Mode” on page 385](#)

Domain Mode

In domain mode, you configure the PFS Steelhead appliance to join a Windows domain (typically, your company domain). When you configure the Steelhead appliance to join a Windows domain, you do not have to manage local accounts in the branch office, as you do in Local Workgroup mode.

Domain mode allows a DC to authenticate users accessing its file shares. The DC can be located at the remote site or over the WAN at the main data center. The Steelhead appliance must be configured as a Member Server in the Windows 2000, or later, ADS domain. Domain users are allowed to access the PFS shares based on the access permission settings provided for each user.

Data volumes at the data center are configured explicitly on the proxy file server and are served locally by the Steelhead appliance. As part of the configuration, the data volume and ACLs from the origin server are copied to the Steelhead appliance. PFS allocates a portion of the RiOS data store for users to access as a network file system.

Before you enable domain mode in PFS:

- configure the Steelhead appliance to use NTP to synchronize the time. For details, see the *Steelhead Appliance Management Console User’s Guide*.

- configure the DNS server correctly. The configured DNS server must be the same DNS server to which all the Windows client machines point.
- have a fully-qualified domain name for which PFS is configured. This domain name must be the domain name for which all the Windows desktop machines are configured.
- set the owner of all files and folders in all remote paths to a domain account and not to a local account.

When you are in domain mode, PFS does not support local user and group accounts. These accounts reside only on the host where they are created. During an initial copy from the origin file server to the PFS Steelhead appliance, if PFS encounters a file or folder with permissions for both domain and local accounts, the Steelhead appliance preserves only the domain account permissions. If your DC is across the WAN, in the event of a WAN outage, you cannot perform user authentication. To prevent this, you either need a local DC, or you can switch to Local Workgroup mode, which requires you to configure local usernames and passwords or use shares that are open to everyone.

For details, see [“Local Workgroup Mode” on page 385](#).

Regarding the user account required to join the Steelhead appliance to the domain:

- This account does *not* need to be a domain admin. Any account that has sufficient privileges to join a machine to Active Directory works (that is; if you have created a non-domain Admin account that has permission to add machines accounts, and it works for regular Windows computers).
- Regardless of what account is entered, RiOS does *not* store the account on the Steelhead appliance. RiOS uses it for a one-time attempt to join the domain.
- If you ever need to rejoin the computer (for example, if the account was deleted from the Active Directory), you must re-enter your credentials.

For details on the how ACLs are propagated from the origin server to a PFS share, refer to the Riverbed Support site at <https://support.riverbed.com>.

Local Workgroup Mode

In Local Workgroup mode you define a workgroup and add individual users that have access to the PFS shares on the Steelhead appliance.

Use Local Workgroup mode in environments where you do not want the Steelhead appliance to be a part of a Windows domain.

Note: If you use Local Workgroup mode, you must manage the accounts and permissions for the branch office on the Steelhead appliance. The local workgroup account permissions might not match the permissions on the origin file server.

PFS Share Operating Modes

PFS provides Windows file service in the Steelhead appliance at a remote site. When you configure PFS, you specify an operating mode for each individual file share on the Steelhead appliance. The proxy file server can export data volumes in Broadcast mode, Local mode, and Stand-Alone mode. After the Steelhead appliance receives the initial copy of the data and ACLs, shares can be made available to local clients. In Broadcast and Local mode only, shares on the Steelhead appliance are periodically synchronized with the origin server at intervals you specify, or manually if you choose. During the synchronization process, the Steelhead appliance optimizes this traffic across the WAN. The following modes are available:

- **Broadcast mode** - Use Broadcast mode for environments seeking to broadcast a set of read-only files to many users at different sites. Broadcast mode quickly transmits a read-only copy of the files from the origin server to your remote offices.

The PFS share on the Steelhead appliance contains read-only copies of files on the origin server. The PFS share is synchronized from the origin server according to parameters you specify when you configure it. Incremental synchronization tries to fetch modified data from origin server, but some changes might not be propagated. Performing full synchronization always synchronizes all data.

- **Local mode** - Use Local mode for environments that must efficiently and transparently copy data created at a remote site to a central data center, perhaps where tape archival resources are available to back up the data. Local mode enables read-write access at remote offices to update files on the origin file server.

After the PFS share on the Steelhead appliance receives the initial copy from the origin server, the PFS share copy of the data becomes the master copy. New data generated by clients is synchronized from the Steelhead appliance copy to the origin server based on parameters you specify when you configure the share. The folder on the origin server essentially becomes a backup folder of the share on the Steelhead appliance. If you use Local mode, users must not directly write to the corresponding folder on the origin server.

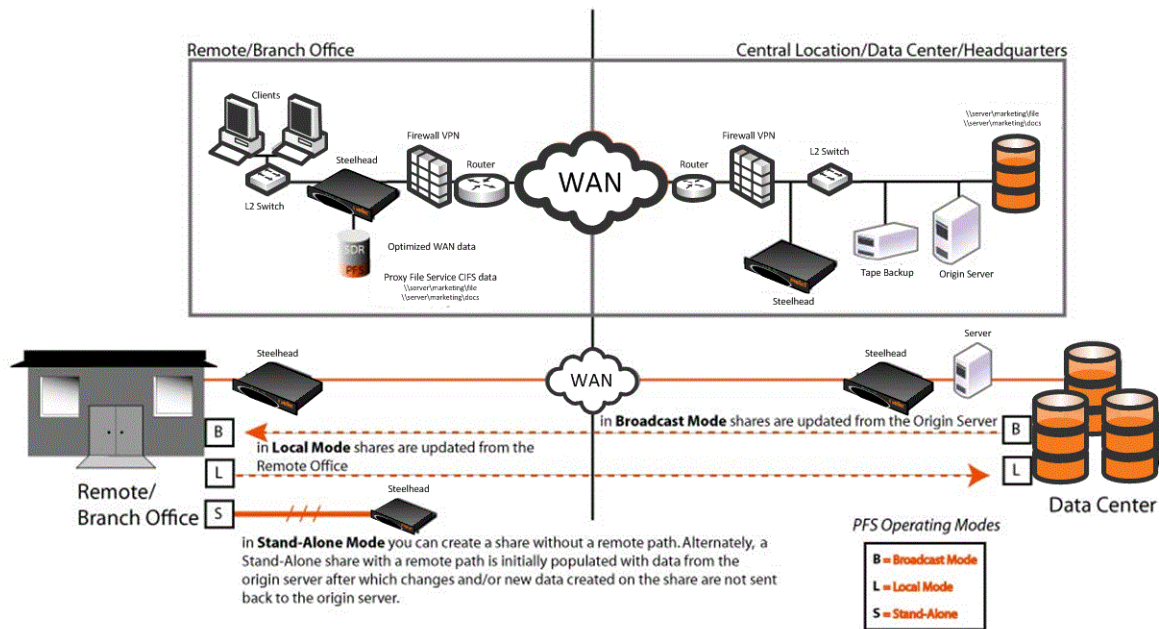
Important: In Local mode, the Steelhead appliance copy of the data is the master copy; do not make changes to the shared files from the origin server while in Local mode. Changes are propagated from the remote office hosting the share to the origin server.

Important: Riverbed recommends that you do not use Windows file shortcuts if you use PFS. For detailed information, contact Riverbed Support site at <https://support.riverbed.com>.

- **Stand-Alone mode** - Use Stand-Alone mode for network environments where it is more effective to maintain a separate copy of files that are accessed locally by the clients at the remote site. The PFS share also creates additional storage space.

The PFS share on the Steelhead appliance is a one-time, working copy of data copied from the origin server. You can specify a remote path to a directory on the origin server, creating a copy at the branch office. Users at the branch office can read from or write to stand-alone shares but there is no synchronization back to the origin server because a stand-alone share is an initial and one-time only synchronization.

Figure 18-1. PFS Deployment



Important: If you set up a PFS share on a NetApp filer, the filer allows all users access regardless of the permissions set on the NetApp share. For example, if you set "No Access" for a user for a share, the NetApp filer does not translate it into the appropriate ACL (Access Control List) entry on the folder. When a PFS share is created from this origin share, the user is allowed access to the share because there is not a deny entry present in the ACL.

Lock Files

When you configure a v3.x Local mode share or any v2.x share (except a Stand-Alone share in which you do not specify a remote path to a directory on the origin server), a text file (`._rbt_share_lock.txt`) that keeps track of which Steelhead appliance owns the share is created on the origin server. Do not remove this file. If you remove the `._rbt_share_lock.txt` file on the origin file server, PFS does not function properly. V3.x Broadcast and Stand-Alone shares do not create these files.

Configuring PFS

The following section describes Steelhead appliance requirements for configuring PFS, and basic steps for configuring PFS shares using the Management Console.

The following are prerequisites and tips for using PFS:

- Before you enable PFS, configure the Steelhead appliance to use NTP to synchronize the time. To use PFS, the Steelhead appliance and DC clocks must be synchronized. For details, see the *Steelhead Appliance Management Console User's Guide*.
- The PFS Steelhead appliance must run the same version of the Steelhead appliance software as the server-side Steelhead appliance.
- PFS traffic to and from the Steelhead appliance travels through the Primary interface. For details, see the *Steelhead Appliance Installation and Configuration Guide*.
- The PFS share and origin-server share names cannot contain Unicode characters; the Management Console does not support Unicode characters.
- Ensure that the name of the Steelhead appliance is entered into your DNS server, and that a host record exists for it. The Steelhead appliance name must either resolve to your Primary or your Auxiliary interface. Failure to resolve the Steelhead appliance name results in an inability to join a Windows 2000 or 2003 domain.

Perform the following basic steps on the client-side Steelhead appliance to configure PFS.

For the server-side Steelhead appliance, you need only verify that it is intercepting and optimizing connections. No configuration is required for the server-side Steelhead appliance.

To configure basic PFS

1. Configure the Steelhead appliance to use NTP to synchronize the time in the Management Console.
2. Choose Configure > Branch Services > PFS Settings page:
 - Enable PFS.
 - Restart the optimization service.
 - Configure either domain or local workgroup settings, as described in [“Domain and Local Workgroup Settings” on page 384](#).
 - If you configured domain settings, join a domain. If you configured local workgroup settings, join a workgroup.

Note: To join a domain, the Windows domain account must have the correct privileges to perform a **join domain** operation.

- Start PFS.
 - Optionally, configure additional PFS settings, such as security signature settings, the number of minutes after which to time-out idle connections, and the local administrator password.
3. Create and manage PFS shares in the Configure > Branch Services > PFS Shares page.
 4. Configure PFS share details in the Configure > Branch Services > PFS Shares Details page:
 - Enable and synchronize PFS shares.
 - If you have v2.x PFS shares (created by Steelhead appliance software v2.x), upgrade them to v3.x shares. By default, Steelhead appliance software v3.x or later creates v3.x shares, which you do not need to upgrade.

- Optionally, modify PFS share settings.
- Optionally, perform manual actions such as full synchronization, cancelling an operation, and deleting shares.

For details, see the *Steelhead Appliance Management Console User's Guide*.

CHAPTER 19 Video Optimization

This chapter describes how to optimize video across your network. Video is one of the fastest growing technologies in business today. Video supports an increasing number of applications, business processes, and training videos. Video on the WAN frequently consumes 30-60 percent of the bandwidth. Whether as video-heavy applications or streaming media, video can make up over half of traffic.

This chapter includes the following sections:

- [“Overview of Video Optimization” on page 391](#)
- [“HTTP Stream Splitting” on page 392](#)
- [“Video On-Demand with HTTP Prepopulation” on page 395](#)

Overview of Video Optimization

You can use video to disseminate information either through live or recorded content. As of RiOS v7.0 there are two solutions for video optimization:

- **Live Streaming with split-streaming** - *Live* streams are only available at a specific time. Examples include video streams of a live sporting event or broadcasting executive events to the workforce. In RiOS v7.0, you can optimize live streaming video with HTTP stream splitting.

For more information about HTTP stream splitting see, [“HTTP Stream Splitting” on page 392](#).

- **On-demand with HTTP prepopulation** - *On-demand* streams are stored on a server and transmitted when requested by a user: for example, training videos. In RiOS v7.0 or later, you can optimize pre-recorded video by using HTTP prepopulation.

For more information about on-demand with HTTP prepopulation, see [“Video On-Demand with HTTP Prepopulation” on page 395](#) and the *Steelhead Appliance Deployment Guide - Protocols*.

Distribution of video impacts not only the overall bandwidth use, but also impacts other services running on the network. In the case of live broadcasting, you might need multiple simultaneous streams per broadcast to support multiple bit rates (for example, remote or wireless workers might watch video at a lower bit rate, while viewers in the office might watch video at a higher bit rate for a larger screen). As the number of streams increases, the likelihood increases that other services, such as email, are affected.

Depending on the codec, resolution, and software in use, video distribution can use anywhere between 16 Kbps (for a low resolution, highly compressed video stream optimized for a small display) to 15 Mbps or more (for a high resolution HDTV stream) per stream. Typical enterprise video streams use between 350 and 500 Kbps to support a single user desktop application.

Connectivity can be overwhelmed quickly and does not scale well when you compare these speeds to typical branch office connectivity, in which you send a single stream for each user. For example, a typical T1 circuit running at 1.544 Mbps (or an E1 running at 2.048 Mbps) might be able to serve users' needs for many business applications, but it cannot support more than 4-5 concurrent video streams. When the entire company might be watching a video broadcast or video conference at the same time, the need for efficient delivery of video across the WAN becomes clear.

When you optimize video streams of either type, you reduce the overall impact of video traffic on the WAN, thereby ensuring service continuity and optimal user experience.

HTTP Stream Splitting

RiOS uses HTTP stream splitting to optimize the following different live video technologies:

- Microsoft Silverlight (RiOS v7.0 or later)
- Adobe HTTP Dynamic Streaming (RiOS v7.0 or later)
- Apple HTTP Live Streaming (RiOS v8.5 or later)

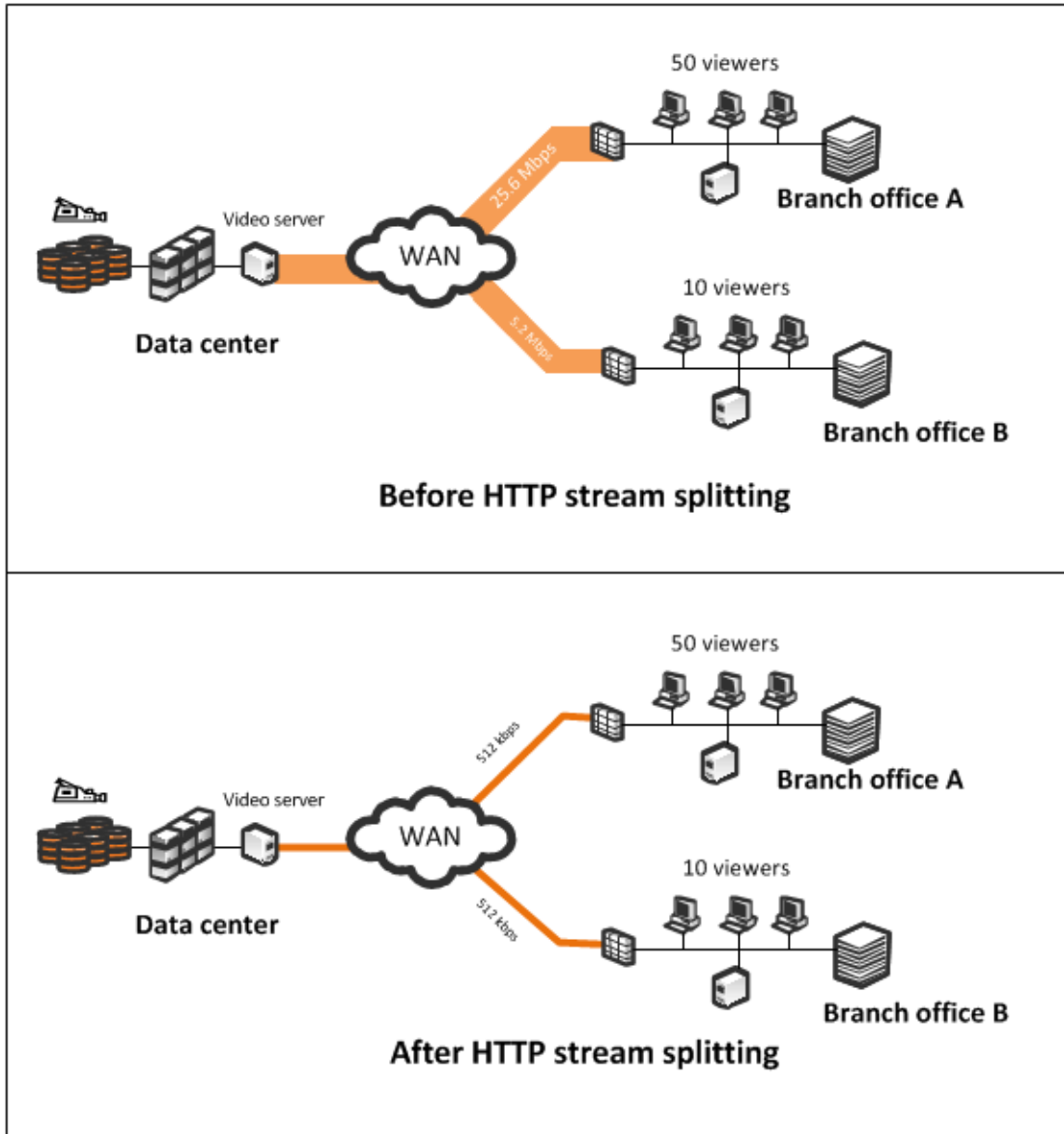
You can optimize video technologies for on-demand video. For details, see [“Video On-Demand with HTTP Prepopulation” on page 395](#).

Note: This section requires you be familiar with your origin server and video encoder.

An unoptimized live video stream can saturate a T1 link with as few as four viewers. Normally, each client that connects to view video draws its own stream, quickly exhausting the resources of smaller branch office links. With stream splitting, one stream is sent from the data center to each branch office, and software at each of the branch offices splits or replicates the stream for each individual client connecting.

You can use HTTP stream splitting to reduce the redundancy of streams operating between the head-end video server and the branch office clients. When you enable stream splitting, the first request for a video stream is sent out over the WAN, and the redundant requests are sent by the Steelhead appliance when the first request is complete. As a result, only one copy of the stream is sent across the WAN no matter how many viewers are tuned in for the live stream.

Figure 19-1. Video Streaming Before and After HTTP Splitting



You can deliver seamless live and on-demand video. By using new streaming technology, you can ensure viewers always get the quality of video best suited for their conditions. Viewers with more bandwidth and processing power receive a higher-quality video stream than viewers with less bandwidth and processing power.

To enable HTTP stream splitting

1. Set up the video origin server.

- On the Steelhead Management Console, open Configure > Optimization > HTTP.

Figure 19-2. Microsoft Silverlight Stream Splitting on the HTTP Page

Configure > Optimization > HTTP ?

Settings

☒ Enable HTTP Optimization

Object Prefetch Table Settings:

☒ Store All Allowable Objects

☐ Store Objects With The Following Extensions:

☐ Disable The Object Prefetch Table

Minimum Object Prefetch Table Time: seconds

Maximum Object Prefetch Table Time: seconds

Extensions to Prefetch:

☒ Enable HTTP Stream Splitting

☒ Enable Per-Host Auto Configuration

<p>Basic Tuning</p> <p><input checked="" type="checkbox"/> Strip Compression</p> <p><input checked="" type="checkbox"/> Insert Cookie</p> <p><input checked="" type="checkbox"/> Insert Keep-Alive</p> <p>Prefetch Schemes</p> <p><input checked="" type="checkbox"/> URL Learning</p> <p><input checked="" type="checkbox"/> Parse and Prefetch</p> <p><input checked="" type="checkbox"/> Object Prefetch Table</p>	<p>Authentication Tuning</p> <p><input checked="" type="checkbox"/> Reuse Auth</p> <p><input checked="" type="checkbox"/> Force NTLM</p> <p><input checked="" type="checkbox"/> Strip Auth Header</p> <p><input checked="" type="checkbox"/> Gratuitous 401</p> <p>SharePoint</p> <p><input type="checkbox"/> FPSE</p> <p><input type="checkbox"/> WebDAV</p>
---	---

☐ Enable Kerberos Authentication Support

Note: The server-side appliance must be joined to the [Windows Domain](#) in order to use this feature. Additionally, NTLM Delegation Mode and Kerberos Authentication require configuration of [Windows Domain Authentication](#) on the server-side appliance.

- Select Enable HTTP Stream Splitting.

Your video is now automatically optimized.

For CLI commands associated with this feature, see the *Riverbed Command-Line Interface Reference Manual*.

The following resources provide more information:

- For details on Microsoft Silverlight smooth streaming, go to <http://www.silverlight.net/>.
- For more information about Adobe dynamic HTTP streaming, go to http://help.adobe.com/en_US/flashmediaserver/devguide/WSC1835B89-E6FF-46cd-AC7D-0E7A8EB331DDDev.html.
- For more information about Apple HTTP live streaming, go to http://developer.apple.com/library/ios/#technotes/tn2224/_index.html.
- For more information about video solutions, see the white paper *Video Architectures with Riverbed*.

Video On-Demand with HTTP Prepopulation

Company updates and new internal training videos are examples of video content that can cause bursts of traffic when they are first made available. Video content is generally accessible only from Web servers and can only be accessed using HTTP. Prewarming RiOS data store during off hours helps to reduce WAN bandwidth consumption during peak hours. While HTTP prepopulation enables you to prepopulate any information over HTTP, this feature is geared toward video optimization.

For more information about HTTP prepopulation, see *Steelhead Appliance Deployment Guide - Protocols*.

CHAPTER 20 Authentication, Security, Operations, and Monitoring

This chapter describes how to configure RADIUS or TACACS+ authentication for the Steelhead appliance, including best practices for securing the Steelhead appliance, and provides information about operations and flow data monitoring.

This chapter includes the following sections:

- [“Overview of Authentication” on page 397](#)
- [“Authentication Features” on page 398](#)
- [“Configuring a RADIUS Server” on page 399](#)
- [“Configuring a TACACS+ Server” on page 402](#)
- [“Securing Steelhead Appliances” on page 403](#)
- [“REST API Access” on page 415](#)
- [“Capacity Planning” on page 416](#)
- [“Overview of Exporting Flow Data” on page 419](#)
- [“SNMP Monitoring” on page 420](#)
- [“Configuring SNMP v3 Authentication and Privacy” on page 426](#)

Overview of Authentication

You can log into a Steelhead appliance with a RADIUS or TACACS+ authentication system for administrative and monitoring purposes. The following methods for user authentication are provided with the Steelhead appliance:

- Local
- RADIUS
- TACACS+

For details on per-command authorization and per-command accounting, see the *Riverbed Command-Line Interface Reference Manual*.

The order in which authentication is attempted is based on the order specified in the AAA method list. The authentication list provides backup authentication methods in case one method fails to authenticate the server. If the first server is unavailable, the next server in the list is contacted depending on the RADIUS/TACACS+ settings.

If there are multiple servers within a method (assuming the method is contacting authentication servers) and a server time-out is encountered, the next server in the list is tried. If the current server being contacted issues an authentication reject, another server is contacted according to the RADIUS/TACACS+ setting. If none of the methods validate a user, the user is not allowed access to the server.

The Steelhead appliance does not have the ability to set a per interface authentication policy. The same default authentication method list is used for all interfaces. You cannot configure authentication methods with subsets of the RADIUS or TACACS+ servers specified (that is, there are no server groups).

For details on Windows domain authentication for encrypted MAPI and SMB signed CIFS traffic see the *Steelhead Appliance Deployment Guide - Protocols*.

Authentication Features

RiOS v5.0.x or later supports the following features (available only through the CLI):

- **Per-command Authorization** - Your TACACS+ server can authorize all CLI commands with the **aaa authorization per-command default** command. The methods available for per-command authorization are local (default) and TACACS+.

To use TACACS+ for per-command authorization, configure the Steelhead appliance for TACACS+ and define the users and commands authorized to execute on your TACACS+ server. For details on how to configure your TACACS+ server, see the TACACS+ server documentation.

Per-command authorization applies to the CLI only.

If you do not have a TACACS+ server, use role-based accounts locally on the Steelhead appliance to limit the Management Console and CLI commands available to users.

For more details on configuring TACACS+ on the Steelhead appliance see [“Configuring a TACACS+ Server” on page 402](#). For details, see how to restrict user roles on [“Best Practices for Securing Access to Steelhead Appliances” on page 404](#).

- **Per-command Accounting** - You always enable per-command accounting locally. You must specifically enable the command for TACACS+ by defining the TACACS+ method using the **aaa accounting per-command default** command. TACACS+ per-command accounting is always sent to all the configured TACACS+ servers. The local method logs the command in the system logs.
- **TACACS+ Server First Hit** - When the first server hit CLI command (**tacacs-server first-hit**) is enabled the Steelhead appliance rejects authentication after the first rejection received from a TACACS+ server rather than continuing through all the TACACS+ servers in the list. This feature applies to user authentication and per-command authorization.
- **Fallback** - The *fallback* option decides how the successive authentication methods are tried. When you enable fallback, if authentication fails, the system continues through all authentication methods (TACACS+, RADIUS, local) in the order you configure them in the authentication method list. Fallback is enabled by default. When you enable conditional fallback (**aaa authentication cond-fallback**) you can configure the system to only proceed beyond TACACS+ or RADIUS if the servers are unreachable. Conditional fallback enables you to reject the login once the first method rejects the attempt, instead of proceeding to the next method in the authentication method list.

- **Remote and Console Method Lists** - There are two method lists: remote (ssh, Web UI) and console (serial, terminal, Steelhead appliance, telnet). The console method requires a local method to be present but the remote list does not. You enable the remote method using the **aaa authentication login default** command. You enable the console method using the **aaa authentication console-login default** command.

Configuring a RADIUS Server

This section describes how to configure a RADIUS server for the Steelhead appliance. This section includes the following topics:

- [“Configuring a RADIUS Server with FreeRADIUS” on page 399](#)
- [“Configuring RADIUS Authentication in the Steelhead Appliance” on page 400](#)
- [“Configuring RADIUS CHAP Authentication” on page 401](#)

Configuring a RADIUS Server with FreeRADIUS

On a per-user basis, you can specify a different local account mapping by using a vendor specific attribute. This section describes how to configure the FreeRADIUS server to return an attribute (which specifies the local user account as an ASCII string). The file paths are the default values. If the RADIUS server installation has been customized, the paths might differ.

Dictionary files are stored in the directory `/usr/local/share/freeradius`. You can define RADIUS attributes in this directory. Assuming the vendor does not have an established dictionary file in the FreeRADIUS distribution, begin the process by creating a file called: `dictionary.<vendor>` in this directory.

The contents of the `dictionary.<vendor>` file define a vendor identifier (which ought to be the Structure of Management Information (SMI) Network Management Private Enterprise Code of the Vendor), and the definitions for any vendor specific attributes.

In the following example, the Vendor Enterprise Number for Riverbed is 17163 and the Enterprise Local User Name Attribute is 1. These numbers specify that a given user is an admin or monitor user in the RADIUS server (instead of using the Steelhead appliance default for users not named admin and monitor).

These instructions assume you are running FreeRADIUS, v.1.0, which is available from <http://www.freeradius.org>. You can also find more details in the *Steelhead Appliance Management Console User's Guide*.

To install FreeRADIUS on a Linux computer

1. Download FreeRADIUS from <http://www.freeradius.org>.
2. At your system prompt, enter the following commands:

```
tar xvzf freeradius-$VERSION.tar.gz
cd freeradius-$VERSION
./configure
make
make install #as root
```

To add acceptance requests on the RADIUS server

1. In a text editor, open the `/usr/local/etc/raddb/clients.conf` file.

2. To create the key for the RADIUS server, add the following text to the `clients.conf` file:

```
client 10.0.0.0/16 {
    secret = testradius
    shortname = main-network
    nastype = other
}
```

The secret you specify here must also be specified in the Steelhead appliance when you set up RADIUS server support.

3. In a text editor, create a `/usr/local/share/freeradius/dictionary.rbt` file for Riverbed.
4. Add the following text to the `dictionary.rbt` file.

```
VENDOR      RBT      17163
ATTRIBUTE    Local-User      1      string      RBT
```

5. Add the following line to the `/usr/local/share/freeradius/dictionary`:

```
$INCLUDE dictionary.rbt
```

6. Add users to the RADIUS server by editing the `/usr/local/etc/raddb/users` file, for example:

```
"admin"      Auth-Type := Local, User-Password == "radadmin"
              Reply-Message = "Hello, %u"
"monitor"    Auth-Type := Local, User-Password == "radmonitor"
              Reply-Message = "Hello, %u"
"raduser"    Auth-Type := Local, User-Password == "radpass"
              Local-User = "monitor", Reply-Message = "Hello, %u"
```

7. Start the server using `/usr/local/sbin/radiusd`. Use the `-X` option if you want to debug the server.

Note: The **raduser** is the monitor user as specified by Local, User-Password.

Configuring RADIUS Authentication in the Steelhead Appliance

The following describes the basic steps for configuring RADIUS authentication in the Steelhead appliance. For details, see the *Steelhead Appliance Installation and Configuration Guide* and the *Steelhead Appliance Management Console User's Guide*.

You prioritize RADIUS authentication methods for the system and set the authorization policy and default user.

Important: Make sure to put the authentication methods in the order in which you want authentication to occur. If authorization fails on the first method, the next method is attempted, and is continued until all the methods have been attempted.

Perform the following basic steps to configure RADIUS support.

To configure RADIUS support

1. Add the IP address of the RADIUS server and specify the key used when you added the device to the ACS server:

```
(config)# radius-server host 192.168.1.200 key rvbd
```

2. Enable AAA.
3. Define the authentication method.

The following configuration attempts to use RADIUS and then local:

```
(config)# aaa authentication login default radius local
```

Configuring RADIUS CHAP Authentication

In RiOS v8.0 or later you can configure RADIUS CHAP authentication through the CLI; or in RiOS v8.5 or later, the Steelhead Management Console. Riverbed does not recommend that you configure CHAP or PAP authentication for RADIUS. Choose which method to use based on the appropriate risk mitigation strategy provided by either option. For example, CHAP transmits the password in a more secure manner, but various RADIUS servers can store the password in an unencrypted format.

```
radius-server host 192.168.198.136 auth-type chap timeout 3 retransmit 1 key testradius
```

To configure CHAP authentication in the Steelhead Management Console, configure the RADIUS server (Configure > Security > RADIUS).

Figure 20-1. RADIUS CHAP Authentication

RADIUS Servers:

▼ Add a RADIUS Server — Remove Selected

Hostname or IP Address: 192.168.198.136

Authentication Port: 1812

Authentication Type: ☐ PAP ☒ CHAP

☒ Override the Global Default Key

Server Key:

Confirm Server Key:

Timeout (seconds): 3 (1 - 60)

Retries: 1 (0 - 5)

☒ Enabled

Add

After you add the server, include RADIUS in the order of authentication methods. A best practice to ensure that you can still perform authentication in the absence of the RADIUS server is to:

- Use the RADIUS server first for authentication, but

- Fall back to the Steelhead appliance username and password database if the RADIUS server is unavailable

Figure 20-2. RADIUS Authentication

Configure > Security > General Security Settings ?

Authentication Methods

RADIUS; Local

☒ For RADIUS/TACACS+, fallback only when servers are unavailable

Authorization Policy: Remote First

Default User: admin

Apply

Configuring a TACACS+ Server

This section describes how to configure a TACACS+ server for the Steelhead appliance. This section includes the following topics:

- [“Configuring TACACS+ with Cisco Secure Access Control Servers” on page 402](#)
- [“Configuring TACACS+ Authentication in the Steelhead Appliance” on page 402](#)

Configuring TACACS+ with Cisco Secure Access Control Servers

The following section requires that you are running a Cisco Secure Access Control Server (ACS) and you want to configure it for TACACS+.

The TACACS+ Local User Service is **rbt-exec**. The Local User Name Attribute is **local-user-name**. This attribute controls whether a user who is not named **admin** or **monitor** is an administrator or monitor user (instead of using the Steelhead appliance default value). For the Steelhead appliance, the users listed in the TACACS+ server must have PAP authentication enabled.

Use the following procedures configure TACACS+ with Cisco Secure ACS.

- To configure TACACS+ with Cisco ACS 4.x, see <http://supportkb.riverbed.com/support/index?page=content&id=S14831>.
- To configure TACACS+ with Cisco ACS 5.x, see <http://supportkb.riverbed.com/support/index?page=content&id=S16158>.

Configuring TACACS+ Authentication in the Steelhead Appliance

The following describes the basic steps for configuring TACACS+ authentication in the Steelhead appliance. You prioritize TACACS+ authentication methods for the system and set the authorization policy and default user.

For more information and detailed procedures, see the *Steelhead Appliance Installation and Configuration Guide* and the *Steelhead Appliance Management Console User's Guide*.

Important: Make sure to put the authentication methods in the order in which you want authentication to occur. If authorization fails on the first method, the next method is attempted, and is continued until all the methods have been attempted.

Perform the following basic steps to configure TACACS+ support.

To configure TACACS+ support

1. Add the IP address of the ACS server and specify the key used when you added the device to the ACS server.

```
(config)# tacacs-server host 192.168.1.200 key rvbd
```

2. Enable AAA.

3. Define the authentication method.

The following configuration attempts to use TACACS+ and then local:

```
(config)# aaa authentication login default tacacs+ local
```

Securing Steelhead Appliances

This section describes security features you can use to harden your network, including ways to secure the Steelhead appliances and some common sense security policies. This section includes the following topics:

- [“Overview of Securing Steelhead Appliances” on page 403](#)
- [“Best Practices for Securing Access to Steelhead Appliances” on page 404](#)
- [“Best Practices for Enabling Steelhead Appliance Security Features” on page 410](#)
- [“Best Practices for Policy Controls” on page 413](#)
- [“Best Practices for Security Monitoring” on page 413](#)
- [“Configuring SSL Certificates for Web User Interface” on page 414](#)

Overview of Securing Steelhead Appliances

In the past, organizations have focused attention on securing their networks by providing security for and preventing attacks against hosts. Unfortunately, there are also many security risks associated with networking devices. Attacks against such devices can be used to gather valuable information. For example, an attacker could use tools to fill up the MAC address tables of Ethernet switches, causing the switches to flood packets. These packets might contain passwords that can easily be captured.

The Steelhead appliance has been certified and subsequently deployed for internal use by several highly security-conscious organizations, including military, government, and financial organizations. However, Steelhead appliances are complex network-facing systems and must be treated accordingly.

Important: Because security requirements vary by organization, consider these recommendations with your particular security goals in mind. Before implementing any security measure described in this section, you must have a thorough understanding of its impact. For example, you do not want to disable access to a Steelhead appliance by mistake and not be able to undo the change because you inadvertently blocked your own access.

If you have a specific security concern, Riverbed recommends that you consult with Riverbed Professional Services.

Best Practices for Securing Access to Steelhead Appliances

This section describes best practices for securing access to your Steelhead appliances. These practices are not requirements, but Riverbed recommends that you consider these suggestions as implementing them can enforce a secure deployment:

- **Restrict physical access** - You must restrict physical access to any network device. An unauthorized user can easily gain access to a Steelhead appliance if that person has physical access. Every device has the ability to recover lost passwords. By acquiring physical access to a device, an attacker can gain control by using the lost password recovery procedures. Even without breaking into the Steelhead appliance software, it is possible to gain access to the contents of disks by gaining access to the Steelhead appliance itself. You should treat the Steelhead appliance as comparable in value to the servers or clients that hold sensitive data. For example, if servers are in locked rooms with armed guards, Riverbed recommends the Steelhead appliances also be in locked rooms.

Another issue with allowing physical access is that it is possible for someone to remove the Steelhead appliance without authorization, allowing an attacker to gain access to confidential data. In general, Steelhead appliances are less valuable to an attacker than application servers or file servers because of an intrinsic scrambling of the RiOS data store. Steelhead appliances also support encryption of the RiOS data store to further reduce the likelihood of a successful attack, and Steelhead Mobile likewise allows the use of file encryption for the RiOS data store on a Window PC.

A third issue related to allowing physical access is the increased susceptibility of the networking device to denial-of-service attacks. A disgruntled employee could conceivably power the Steelhead appliance down, disarrange the cabling, swap hard drives, or even steal the Steelhead appliance.

- **Use an appropriate login message** - The login message appears on the Management Console Home page. You must display a login message that reinforces your organization access and security policies. Have your organization legal council approve a more appropriate login message.

Typical login messages include, but are not limited to:

- statements pertaining to authorized access only
- consequences of unauthorized access
- elimination of right to privacy
- acknowledgment that they might be monitored

The default login message is "Welcome to the Management Console for Steelhead_name!" You can change this by navigating to the Configure > System Settings > Announcements page in the Management Console and specifying another message. Or, you can use the CLI, as shown in the following example.

Syntax:

```
[no] banner login <message string>
```

Example:

```
banner login. "This computer system is the property of Company XYZ Inc. Disconnect NOW if you
have not been expressly authorized to use this system. Unauthorized use is a criminal offence
under the Computer Misuse Act 1990.
Communications on or through Company XYZ Inc. computer systems may be monitored or recorded to
secure effective system operation and for other lawful purposes."
```

- **Allow management only from the primary interface** - Limiting SSH and HTTPS access to the Primary interface allows administrators to restrict who can access the Steelhead appliances by the use of filters or Access Control Lists. These filters are typically based on the source IP addresses of hosts and are applied on network devices like routers, Layer-3 switches, or firewalls. Limiting remote management access to Steelhead appliances helps prevent unauthorized user access.

Syntax:

```
[no] web httpd listen enable
[no] web httpd listen interface <interface>
[no] ssh server listen enable
[no] ssh server listen interface <interface>
```

Example:

```
web httpd listen enable
web httpd listen interface primary
ssh server listen enable
ssh server listen interface primary
```

- **Use SSH version 2** - SSH version 2 is more secure than previous versions of SSH. The major differences between SSH1 and SSH2 fall into two main categories: technical and licensing. Technically speaking, SSH2 uses different encryption and authentication algorithms.

SSH1 offers four encryption algorithms (DES, 3DES, IDEA and Blowfish), because SSH2 dropped support for DES and IDEA, but added three algorithms. SSH1 also used the RSA authentication algorithm, because SSH2 switched to the Digital Signature Algorithm (DSA). These changes were designed to increase the base level of security in SSH2 by using stronger algorithms.

Syntax:

```
[no] ssh server v2-only enable
```

Example:

```
ssh server v2-only enable
```

- **Disable unencrypted communication protocols such as Telnet and HTTP** - An attacker can easily gain access to user names and passwords by sniffing network communications. You might consider a switched Ethernet environment secure, because packets are only forwarded out ports based on the destination MAC address; however, this is not necessarily the case.

Several hacking tools are available that can generate large amounts of bogus MAC addresses. These packets flood the switch's MAC address table in an attempt to overflow the table. A switch typically floods packets out all ports if it does not have an entry in its MAC address table. Therefore, after the MAC address table for the switch is filled, the switch floods packets out all ports.

The attacker can now use a packet-capturing application to capture the flooded packets. They can then look for remote management connections. After the attacker discovers remote management connections, they can reset those TCP connections, causing the user to log in again allowing them to capture the user name and password.

If only HTTPS and SSH are used, the attacker cannot obtain the user names and passwords because they are encrypted.

Syntax:

```
[no] telnet-server enable
[no] web http enable
```

Example:

```
no telnet-server enable
no web http enable
```

In RiOS v8.5 or later, you can configure the Steelhead appliance to redirect HTTP access to HTTPS access with the **web http redirect** command. Riverbed recommends this method unless your security policy requires the Steelhead appliance to not listen to HTTP.

- **Use TLS only for the Management Console** - Only permit TLS between the browser and the Management Console.

Syntax:

```
web ssl protocol tlsv1
no web ssl protocol sslv3
web ssl protocol tlsv1.1
web ssl protocol tlsv1.2
```

- **Restrict user roles** - Be sure to restrict the roles of users. For example, if a Help desk administrator is supposed to only view statistics and generate reports, their account restricts them to those roles.

Syntax:

```
[no] rbm user <username> role <role> permissions <permissions>
[no] rbm role <role> primitive <primitive>
```

Example:

Refer to the *Riverbed Command-Line Interface Reference Manual* for more details on this command.

- **Remove the default user name from the Web preference settings** - The default user name in the login field is admin. Do not display a default user name because it gives an attacker an example of a user name against which to wage a brute-force password attack. Brute-force attacks typically go through an extensive list of words (for example, a dictionary attack) in an attempt to guess the password.

Syntax:

```
web prefs login default
```

Example:

```
web prefs login default ""
```

- **Change all default passwords and community strings** - Be sure to change the default password for the administrator and monitor accounts. The monitor account is disabled by default, unless the Steelhead appliance is upgraded from an older release where the monitor account was enabled.

The most common problem with SNMP is that it uses the default community string of public. Change the default to something different.

Syntax:

```
username <userid> password 0 <cleartext>
```

Example:

```
username admin password 0 o2fMu5TS!
```

Syntax:

```
snmp-server community
snmp trap-community
```


Example:

```
snmp-server community o2fMu5TS!
```

- **Use strong passwords** - Strong passwords typically include combinations of letters, numbers, special characters and combinations of upper and lower case with at least eight characters in length. Strong passwords reduce the likelihood of a successful brute-force attack because they are not found in dictionaries and exponentially increase the complexity of the passwords.

An example of a strong password is o2fMu5TS!

- **Use AAA authentication** - One of the challenges with using local user names and passwords is that when an employee leaves an organization, an administrator must touch every device that has a user name and password configured for that former employee.

By leveraging TACACS+, you gain the advantage of having a single location for configuring user names and passwords. When a person leaves your organization, you can simply disable that single account thereby preventing the user from access to all of the network devices configured to use TACACS+. Another benefit of TACACS+ is the ability to lock out an account after several unsuccessful login attempts.

TACACS+ also provides greater reporting capabilities regarding who is accessing which devices at what time. With a global user name and password, you have no idea which administrator actually logged in at a specific time. These reports can be invaluable for tracking network changes and identifying who is making changes. Therefore, it is a critical tool for change management controls.

Refer to *Riverbed Command-Line Interface Reference Manual* and the *Steelhead Appliance Management Console User's Guide* for more detailed information about how to configure AAA.

- **Configure the CLI session time-out** - By default, the Steelhead appliance closes the SSH session to the command line after 15 minutes. You can configure this interval to be more or less with the following command:

Syntax:

```
cli default auto-logout *
```

Example:

```
cli default auto-logout 10
```

This command only affects new SSH sessions. If you want to modify the time-out session only for the current session (and not affect the default settings), use the following command:

Syntax:

```
cli session auto-logout *
```

You can turn off the auto-logout feature with the following command:

```
no cli default auto-logout
```

This command changes both the current and the default settings.

You can display the current auto-logout settings with the following command:

```
show cli
```

- **Use strong SSL ciphers for management communications** - Be sure to use strong encryption ciphers for any HTTPS management communications. The cipher is the key that encrypts management communications to the Steelhead appliance. An attacker could still use the hacking tools to crack the encrypted user name and password if the encryption ciphers are too weak. An example of a weak cipher is only 56 or 64 bits. A strong cipher is greater than 128 bits.

Syntax:

```
web ssl cipher
```

Example:

```
web ssl cipher "HIGH:-aNULL:-kKRB5:-MD5"
```

- **Set an inactivity timer for console, SSH, and HTTPS sessions** - Be sure to set a proper inactivity time-out value for management sessions. Do not set a console inactivity time-out value to 0. This could allow an attacker to take over a previous management session if the previous administrator did not manually log off.

Syntax:

```
[no] web auto-logout <minutes>
[no] cli default auto-logout <minutes>
```

Example:

```
web auto-logout 10
cli default auto-logout 10
```

- **Ensure SNMP is listening on the management interface only** - To prevent unauthorized SNMP access, Riverbed recommends enabling SNMP access on the Primary interface only. This allows administrators to control who can access Steelhead appliances through SNMP by way of using filters applied to routers, Layer-3 switches, or firewalls.

Syntax:

```
[no] snmp-server listen enable
[no] snmp-server listen interface <interface>
```

Example:

```
snmp-server listen enable
snmp-server listen interface Primary
```

- **Enable link state alarms** - Enable the link state alarms, which are disabled by default. This can alert you to any attempt to modify the cabling on the Steelhead appliances by inserting a tap for illegal sniffing functions.

Syntax:

```
[no] stats alarm {<type> <options>}
```

Example:

```
stats alarm linkstate enable
```

- **Disable the auto-discover CMC feature** - By default, all Steelhead appliances try to register with the CMC using the default hostname riverbedcmc. If you do not have a CMC, disable this feature.

If you do have a CMC, Riverbed recommends that you use it to manually discover Steelhead appliances, thereby reducing the possibility that an attacker could compromise the DNS environment and change the IP address of the riverbedcmc 'A' record to a rogue CMC.

Syntax:

```
[no] cmc enable
```

Example:

```
no cmc enable
```

- **Configure a password policy in the Steelhead appliance** - In RiOS v8.0 or later, you can configure user accounts on the Steelhead appliance with a password policy to enforce minimum security standards for passwords, number of password attempts, and password expiration. The following password policies are pre-defined in the Steelhead appliance:

- **Strong security template** - Corresponds to the typical recommendations in Federal password guidelines.
- **Basic security template** - Provides a minimal set of standards.

For more information about the type of accounts, how to configure role-based accounts, and details on the options for the password policy, see the *Steelhead Appliance Management Console User's Guide*.

- **Use a BIOS password** - Enable a password for BIOS, which prevents admin password recovery without the supervisor password. To configure a BIOS password:
 - Connect a null modem cable to a Steelhead appliance.
 - Open up a terminal on your host to the Steelhead appliance.
 - Power up the Steelhead appliance.
 - Press F4 to enter BIOS.
 - Navigate to Security tab.
 - Specify a supervisor password.
 - Make sure that the user password option is set to OFF.
 - Save your configuration and continue to boot the Steelhead appliance.
- **Use a boot loader password** - When you reboot the Steelhead appliance, you can select from one of the two RiOS images on the menu. You can perform password recovery at this point by pressing the E for edit. Pressing E alters the boot sequence to change the administrative password. Using the following commands, you can enable the boot loader to lock the password recovery process until a password is entered.

```
Steelhead (config) # boot bootloader password test1234
Steelhead (config) # write mem
Steelhead (config) # reload
...
```

```
-----
0: Riverbed Steelhead Software v. 5.5.3 (64bit)
1: Riverbed Steelhead Software v. 5.5.3 (64bit)
-----
```

```
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS or 'p' to enter a
password to unlock the next set of features.
```

```
Press 'P'
Password: *****
```

- **SSL Issues with Internet Explorer 6 and Oracle R12** - Previously, RiOS fixed a vulnerability found in CBC-based ciphers prior to versions 0.9.6e by inserting an empty frame on the wire to avoid a Chosen Plaintext Attack on cipher-block chaining (CBC) ciphers. Some versions of client and server applications do not understand the insertion of empty frames into the encrypted stream and close the connection when they detect these frames. Therefore, RiOS no longer inserts empty frames by default. Examples of applications that close the connection when they detect these empty frames are IE6 and Oracle R12. SharePoint under IIS has also exhibited this behavior.

The failure occurs when the SSL application fails to understand the data payload when either the client or server is using a block cipher using CBC mode as the chosen cipher. This can be with DES, AES, or 3DES using CBC. Note that when Steelhead appliances are deployed, the chosen cipher can be different than when the client is negotiating directly with the SSL server.

Important: Because current Web browsers do not protect themselves from this vulnerability, Steelhead appliances are no less secure than other vendor's appliances. From a security perspective, fixing this vulnerability is the responsibility of a server, not a patched client.

To determine whether the Steelhead appliances are inserting empty frames to avoid an attack, capture TCP dumps on the server-side Steelhead LAN interface and look at the Server Hello message that displays the selected cipher. Verify that DES, AES, or 3DES is the cipher. Also, check for the existence of 32-byte length SSL application data (this is the empty frame) on the LAN traces, followed by an SSL Alert.

To change the default and insert empty frames, enter the CLI command **no protocol ssl bug-work-around dnt-insrt-empty**.

Note: For details on the vulnerability, see <http://www.openssl.org/~bodo/tls-cbc.txt>.

Best Practices for Enabling Steelhead Appliance Security Features

The following best practices enable important security features provided by RiOS. These best practices are not requirements, but Riverbed recommends that you follow these suggestions as implementing them can enforce a secure deployment:

- **Use peering rules to control enhanced auto-discovery** - Enhanced auto-discovery is a feature that allows Steelhead appliances or Mobile Clients to discover other Steelhead appliances using TCP options. This feature greatly reduces the complexities and time it takes to deploy Steelhead appliances. It works so seamlessly that it can occasionally have the undesirable effect of peering with Steelhead appliances on the Internet that are not in your organization's management domain.

Another scenario could be that your organization has a decentralized management approach where different business units might make their own purchasing and management decisions. You might not want Steelhead appliances from two or more business units to peer with one another.

In these situations, Riverbed recommends using peering rules. Peering rules determine which connections your Steelhead appliance optimizes connections with, based on the source and destination IP addresses or TCP ports. This lets you deny peering with any unwanted connections. Another option is to create an Accept peering rule for your corporate network that allows peering from your own IP addresses, and denies it otherwise.

Syntax:

```
[no] in-path peering rule {auto | pass | accept} peer <peerip> ssl-capability
{cap | in-cap | nocheck} src <subnet> | dest <subnet> | dest-port <port> rulenum <rulenum>
description <desc>
```

Example:

```
in-path peering rule accept peer xxx.xxx.xxxx.xxxx/xx
```

For more information about using peering rules, see the *Steelhead Appliance Management Console User's Guide*.

- **Enable a secure inner channel between Steelhead appliances when using the SMB-signing proxy feature** - When sharing files, Windows provides the ability to sign CIFS messages to prevent man-in-the-middle attacks. Each CIFS message has a unique signature which prevents the message from being tampered with. This security feature is called SMB signing.

SMB signing is mandatory on all CIFS connections to domain controllers. Therefore, any CIFS connection to a domain controller must use SMB-signed packets.

You can enable the RiOS SMB signing feature on the server-side Steelhead appliances communicating with servers that have SMB signing set to Required. This alleviates latency in file access with CIFS acceleration while maintaining message security signatures. With SMB signing on, the Steelhead appliance optimizes CIFS traffic by providing bandwidth optimization (RiOS SDR and LZ), TCP optimization, and CIFS latency optimization—even when the CIFS messages are signed.

However, because there is no packet signing taking place between the Steelhead appliances for these connections, Riverbed recommends that you configure a secure inner channel to encrypt the traffic between the Steelhead appliances.

- **Enable a secure inner channel between Steelhead appliances when using Exchange 2007 encryption** - Outlook 2007 has encryption enabled by default. The Steelhead appliances are able to decrypt this traffic; however, the connections between the Steelhead appliances are unencrypted by default. Configure a secure inner channel to encrypt all MAPI traffic between the Steelhead appliances.
- **Enable a secure inner channel to encrypt all optimized traffic between Steelhead appliances** - When you enable a secure inner channel, all data between the client-side and the server-side Steelhead appliances is sent over the secure inner channel. You configure the peer Steelhead appliance as SSL peers so that they are trusted entities. The Steelhead appliances authenticate each other by exchanging certificates as part of the encrypted inner-channel setup. After the Steelhead appliances establish the secure inner channel, you can encrypt and optimize all optimized traffic between Steelhead appliances using the channel. The trust between the Steelhead appliances is bi-directional; the client-side Steelhead appliance trusts the server-side Steelhead appliance, and vice versa.
- **Authenticate WCCP service groups** - By default, WCCP peers in a WCCP group do not use authentication when registering. This could allow an attacker to join a WCCP group and potentially cause a denial of service attack. Also an administrator could accidentally misconfigure a router to use a WCCP group that already is in use. Authentication controls would prevent these rogue devices from peering, thereby preventing possible network outages or degradation of performance.

Syntax:

```
[no] wccp service-group <service-id> {routers <routers> | assign-scheme [either | hash | mask]
| src-ip-mask <mask> | dst-ip-mask <mask> | src-port-mask <mask> | dst-port-mask <mask>}
protocol [tcp | icmp] | encap-scheme [either | gre | l2] | dst-ip-mask <mask> flags <flags> |
password <password> | ports <ports> | priority <priority> | weight <weight> | assign-scheme
[either | hash | mask] | src-ip-mask <mask> | dst-ip-mask <mask> | src-portmask <mask> | dst-
port-mask <mask>}
```

Example:

```
wccp service-group 91 routers x.x.x.x password S3cuRity!
```

- **Encrypt the RiOS data store** - RiOS SDR takes all TCP traffic and segments using a rolling data-driven computation. The segmentation produced is not readily predictable without running the computation, so an attacker interested in reconstructing a particular file does not know how many segments are involved or what the file boundaries are within the segments. The segmentation is stable, so that two identical bit sequences produce the same segmentation. Each new segment identified is written to the RiOS data store, because each previously seen segment is reused.

Even though there is inherent security in the obfuscation of the RiOS data store, Riverbed still provides a mechanism for enabling strong encryption of the RiOS data store. Encrypting the RiOS data store significantly limits the exposure of sensitive data in the event an Steelhead appliance is compromised by loss, theft, or other types of security violations. The secured data is impossible for a third party to retrieve.

Syntax:

```
[no] datastore encryption type {NONE | AES_128 | AES_192 | AES_256}
```

Example:

```
datastore encryption type AES_256
```

Next, select Clear the Data Store on Reboot and reboot the Steelhead appliance.

- **Change the secure vault** - The secure vault contains sensitive information from your Steelhead appliance configuration, including SSL private keys and the RiOS data store encryption key. These configuration settings are encrypted on the disk at all times using AES 256-bit encryption.

Initially, the secure vault is keyed with a default password known only to RiOS. This allows the Steelhead appliance to automatically unlock the vault during system startup. You can change the password, but the secure vault does not automatically unlock on start up. To optimize SSL connections or to use RiOS data store encryption, the secure vault must be manually unlocked if the Steelhead appliance is rebooted.

Therefore, Riverbed recommends using this feature only in conjunction with a CMC. The CMC can automatically unlock the Secure Vault when the Steelhead appliance connects to the CMC after a reload.

Syntax:

```
secure vault {[new-password <password> | reset-password <old password> | unlock <password>]}
```

Example:

```
Secure vault unlock o2fMu5TS!
```

- **Disable unused features** - Disable any features that are not in use. For example, MAPI Exchange is on by default. If your organization uses Lotus Notes, Riverbed recommends that you disable Exchange optimization.

Refer to the *Riverbed Command-Line Interface Reference Manual* or the *Steelhead Appliance Management Console User's Guide* for the specific features you might want to disable.

- **Disable automatic email notification** - This feature proactively sends email notification of critical issues on the Steelhead appliance (such as significant alarms and events) to Riverbed Support. Your organization might not want to send these automatic notifications.

Syntax:

```
[no] email autosupport enable
```

Example:

```
no email autosupport enable
```

- **Disable Steelhead reporting** - This feature proactively reports some very basic information back to Riverbed Support once a week. This reporting is initially disabled, however, if the user configures nameserver IP addresses for the Steelhead appliance it is automatically enabled. Your organization might not want to send this report.

Syntax:

```
[no] support uptime-report enable
```

Example:

```
no support uptime-report enable
```

- **Delete the preconfigured NTP servers** - If your organization has NTP configured internally, Riverbed recommends removing the preconfigured NTP servers.

Syntax:

```
[no] ntp server <ip-addr> <cr> | [version <number>]
```

Example:

```
no ntp server 66.187.224.4
```

- **Configure Network Time Protocol (NTP) settings** - Riverbed recommends that you synchronize the Steelhead appliance to an NTP server of your choice. By default, the appliance uses the Riverbed-provided NTP server. Time is a critical function for the Steelhead appliance and other network devices. Networks rely on accurate time determination for managing, securing, planning, and debugging. Tampering with time sources or posing as a rogue time server can lead to critical issues such as network authentication, or less critical issues such as conflicting log message timestamps.

RiOS v8.0 or later supports MD5-based NTP authentication at the CLI, and RiOS v8.5 or later supports both MD5- and SHA-based NTP authentication on the CLI and Management Console.

- **Disable any interfaces not in use** - Be sure to disable any interfaces that are not being used. Examples include the Auxiliary interface and any unused in-path interfaces.

Syntax:

```
[no] interface <interfacename> <options>
```

Example:

```
interface inpath0_1 shutdown
```

For more information about Steelhead appliance security, see the *Steelhead Appliance Management Console User's Guide*.

Best Practices for Policy Controls

This section includes the best practices for implementing secure policy controls:

- **Use the Simple Certificate Enrollment Protocol (SCEP)** - In RiOS v5.5.2 or later, SCEP allows Steelhead appliances to request signed certificates for enrollment and re-enrollment from the certificate server.
- **Use a Certificate Revocation List (CRL)** - In RiOS v5.5.2 or later, Steelhead appliances can download CRL lists that contain revoked certificates from certificate servers through LDAP. Revoked certificates are considered invalid, and are not used by the Steelhead appliance.

For more details on SCEP and CRL, see the *Steelhead Appliance Deployment Guide - Protocols*.

Best Practices for Security Monitoring

After implementing security measures for your organization, Riverbed recommends enabling the following security monitoring features:

- **Enable Logging** - Be sure to enable logging and log to a syslog server. At a minimum, set logging to the notice level to capture failed login attempts. You can also change the default logging facility (CLI only) (system=local0, user=local1, and per-process=local2). Use the CLI command **logging facility user local# system local# perprocess local#** to configure the syslog facility.

Example—A failed login attempt:

```
Apr 13 05:19:49 BRANCH webasd[6004]: [web.NOTICE]: web: Attempt to Authenticate admin
Apr 13 05:19:49 BRANCH webasd(pam_unix)[6004]: authentication failure; logname= uid=0 euid=0
tty= ruser= rhost= user=admin
Apr 13 05:19:49 BRANCH webasd[6004]: [web.NOTICE]: web: Failed to authenticate user admin: You
must provide a valid account name and password.
```

After you enable syslog and log to a server, remember to review the logs daily.

RiOS v6.0 also includes several SNMP traps to notify you of Steelhead appliance configuration changes, successful logins, and system dump initiation. For more information, see the *Steelhead Appliance Management Console User's Guide*.

Syntax:

```
[no] logging <IP addr> [trap <log level>]
```

Example:

```
logging x.x.x.x trap notice
```

- **Email alerts** - Be sure to enable email alerts internally.

Syntax:

```
[no] email mailhub <hostname or IP addr>
[no] email notify events enable
[no] email notify failures enable
[no] email notify events recipient <email addr>
[no] email notify failures recipient <email addr>
```

Example:

```
email mailhub x.x.x.x
email notify events enable
email notify failures enable
email notify events recipient helpdesk@companyxyz.com
email notify failures recipient helpdesk@companyxyz.com
```

Refer to the *Riverbed Command-Line Interface Reference Manual* for more details on configuring email alerts.

- **Register with the Riverbed forums** - Riverbed has several forums which enable you to receive advanced notifications for:
 - general announcements and updates
 - software releases
 - features

To register with Riverbed forums, go to <https://splash.riverbed.com/welcome>.

Configuring SSL Certificates for Web User Interface

Steelhead appliance automatically generate and use a self-signed certificate to provide HTTPS access to the Web UI to manage the appliance.

This is separate from the SSL feature set. Management of SSL certificates for the Web UI pertains to the SSL certificate used by the appliance's Web UI when HTTPS is used.

You can replace the self-signed certificate with one created by the administrator or generated by a 3rd party certificate authority.

To upload a key or certificate

- Connect to the Steelhead CLI and enter the following command:

```
web ssl cert import pem <PEM text>
```

Do not enter more than one certification and more than one key. Because neither is required, you can opt to update only the certificate.

To generate a new certificate for the existing key for use with HTTPS on the Steelhead appliance

- Connect to the Steelhead CLI and enter the following command:

```
web ssl cert update
```

The new certificate is authorized for one year (365 days).

To generate a brand new self-signed certificate and key pair for use with HTTPS management on the Steelhead appliance

- Connect to the Steelhead CLI and enter the following command:

```
web ssl cert generate
```

This overwrites the existing certificate and key pair regardless of whether the previous one was self-signed or user added. This command generates a self-signed certificate that is authorized for one year (365 days).

REST API Access

You enable access to the Riverbed REST API in the Configure > Security > REST API Access page. REST (REpresentational State Transfer) is a framework for API design. REST builds a simple API on top of the HTTP protocol. It is based on generic facilities of the standard HTTP protocol, including the six basic HTTP methods (GET, POST, PUT, DELETE, HEAD, INFO) and the full range of HTTP return codes. You can discover REST APIs by navigating links embedded in the resources provided by the REST API, which follow common encoding and formatting practices.

You can invoke the REST API to enable communication from one Riverbed appliance to another through REST API calls. For example:

- A Cascade Profiler appliance communicating with a Cascade Shark appliance.
- A Cascade Profiler appliance retrieving a QoS configuration from a Steelhead appliance.

For all uses you must preconfigure an access code to authenticate communication between parties and to authorize access to protected resources.

For more information about the Steelhead appliance REST API, see the *Steelhead Appliance Management Console User's Guide* and the *Steelhead Appliance REST API Guide*.

Capacity Planning

This section describes capacity planning for the Steelhead appliance. This section includes the following topics:

- “Model Characteristics” on page 416
- “Admission Control” on page 417

Model Characteristics

This section describes the characteristics of the optimization resources available for the Steelhead appliances. These resources are the primary determining factors for supported WAN capacity, maximum number of optimized TCP connections, and RiOS data store capacity. For example, the amount of hard drive space and RAM determine how large the RiOS data store can be.

This section includes the following topics:

- “TCP Connections” on page 416
- “WAN Capacity Limits” on page 416
- “RiOS Data Store Size” on page 417
- “Disk Performance” on page 417

For more information about individual model specs, go to <http://www.riverbed.com/products-solutions/products/>.

TCP Connections

Each Steelhead appliance model has a maximum number of optimized TCP connections. The larger the Steelhead appliance, the more CPU, memory, and disk resources are available, increasing the amount of supportable connections. This is one of the primary considerations you use for sizing branch offices. Typically, Riverbed recommends a guideline of 5-10 connections per user. Sizing for data center Steelhead appliances must take in account all optimized connections coming into the data center. For planning purposes, use the high end numbers for these types of connections. Large amounts of active connections (connections that are actively transmitting data), such as HTTP, have more impact on the Steelhead appliance resources.

To view the connection history use the **show stats connection <timeframe>** command.

You can see the average and maximum number of connections for the time frame entered. These numbers are useful to determine if the current Steelhead appliance is properly sized for the number of connections. You can also use this command to review when adding additional users or applications that can increase the number of optimized connections.

WAN Capacity Limits

WAN capacity limit is the amount of optimized traffic that is sent outbound from a Steelhead appliance. This is also commonly used for sizing branch office Steelhead appliances. For data center Steelhead appliances, the WAN capacity is a recommendation on how much typical data throughput a Steelhead appliance can process. Model 1050 and below are limited to the rated outbound capacity (outbound after optimization). No rate limit or hard restriction is applied to throughput for a data center Steelhead appliance (2050 and larger), though excessive amounts of certain types of traffic can strain the resources available on the Steelhead appliance.

RiOS Data Store Size

RiOS data store size is the amount of disk space, in GB, available for SDR use. The data center Steelhead appliances use RAID 10 or FTS 7050 for disk redundancy and optimal performance.

Disk Performance

Take disk performance into account for high-end data center deployments. Certain types of data put more load on the disk systems and can be monitored if performance issues are suspected. High throughput data replication deployments typically use dedicated Steelhead appliances.

Monitor the disk systems with the following OID.

OID	Descriptions
1.3.6.1.4.1.17163.1.1.4.0.8	RAID Errors

Use the following Data Store Disk Load report in the Management Console to monitor disk performance.

If the Disk Load report is showing 80-90% for a sustained amount of time, or multiple times a day that coincide with periods of lower performance and average RiOS data store cost greater than 5000, the disk load might be impacting overall performance.

Admission Control

This section describes admission control and This section includes the following topics:

- [“Connection Limits” on page 417](#)
- [“Memory Limits” on page 419](#)

Admission control prevents the Steelhead appliance from processing traffic when overloaded. It also controls the connection count limits. Admission control stops the interception of connections for optimization but still allows the connections to pass through without optimization. Admission control is in one of two states:

- **Flowing** - In the flowing state, connections are intercepted as normal. Every 30 seconds or every 20 connections, admission control re-evaluates whether the system is within limits. If the system exceeds certain limits, admission control moves into the paused state.
- **Paused** - In the paused state, the Steelhead appliance does not intercept connections. The connections currently intercepted continue to be optimized although new connections are passed through. Every 30 seconds or every 20 connections, admission control re-evaluates whether the system falls below certain limits. If so, admission control moves back into the flowing state.

Connection Limits

Each model contains connection limits to limit the total number of connections that is accepted into the system. The connection limits have rising and falling thresholds. The rising threshold is the cutoff limit. While the system is in flowing state, if the connection count rises above this threshold, admission control moves to the paused state. The falling threshold is the enable limit. While the system is in the paused state, until the connection count falls below this threshold, admission control keeps the system in paused state.

Some leeway is given for connection limits before admission control is triggered. The minimum number of additional allowed connections before entering admission control on any model is 10. For example, a 250L with a rating of 30 connection does not enter admission control until it passes 40 connections. The Steelhead appliance does not exit admission control until the number of connections falls back below the rated limit. Using the same example, a 250L entering admission control at 40 connections does not start optimizing new connections again until it is back down below 30 connections (rated limit).

The Steelhead appliance sends out a SNMP alert, called *Admission Control Error*, to the SNMP host that you define. It alerts you that the licensed optimization limit is reached. You can purchase a bigger Steelhead appliance that can take all the optimized TCP connections; or you can limit the type of traffic to be optimized with in-path rules configuration, and ensure maximum optimization benefits are limited only to most critical traffic or specific traffic that is hogging the WAN bandwidth.

Trap and OID	Steelhead Appliance State	Text	Description
admissionConnError (enterprises.17163.1.1.4.0.11)	Control Admission	Admission control connections alarm has been triggered.	The Steelhead appliance has entered admission control due to the number of connections and is unable to handle the amount of connections going over the WAN link. During this event, the Steelhead appliance continues to optimize existing connections, but new connections are passed through without optimization. The alarm clears automatically when the traffic has decreased and no other action is needed.

Riverbed recommends polling the number of optimized connections periodically, so you can take proactive steps before all the optimized TCP connections are consumed. The following table shows the SNMP MIB for the number of optimized connections.

Trap and OID	Steelhead Appliance State	Text	Description
optimizedConnections (enterprises. 17163.1.1.5.2.1.0)		Current total number of optimized connections.	The optimized connections count is the total of half opened, half closed, and established/flowing connections.

Memory Limits

Each Steelhead appliance model contains memory limits to limit the total amount of memory that is used. The memory limits have rising and falling thresholds. The rising threshold is the cutoff limit. While the system is in flowing state, if the memory usage rises above this threshold, admission control moves to the paused state. The falling threshold is the enable limit. When the system is in the paused state, until memory usage falls below this threshold, admission control keeps the system in paused state.

Trap and OID	Steelhead Appliance State	Text	Description
admissionMemError (enterprises.17163.1.1.4.0.10)	Admission Control	Admission control memory alarm has been triggered.	The Steelhead appliance has entered admission control due to memory consumption. The Steelhead appliance is optimizing traffic beyond its rated capability and is unable to handle the amount of traffic passing through the WAN link. During this event, the Steelhead appliance continues to optimize existing connections, but new connections are passed through without optimization. The alarm clears automatically when the traffic has decreased and no other action is needed.

Note: Use the **show admission** command to display the cutoff and enable settings for your Steelhead appliance.

For additional information about admission control, see the Riverbed Knowledge Base article *Understanding Admission Control* at <https://supportkb.riverbed.com/support/index?page=content&id=s15140>.

Overview of Exporting Flow Data

NetFlow and other Flow Data Collectors gather network statistics about network hosts, protocols and ports, peak usage times, and traffic logical paths. The flow data collectors update flow records with information pertaining to each packet traversing the specified network interface.

The flow data components are as follows:

- **Exporter** - When you enable flow data support on a Steelhead appliance, it becomes a flow data Exporter. The Steelhead appliance exports raw flow data records to a flow data collector. You only need one Steelhead appliance with flow data enabled to report flow records.
- **Collector** - A server or appliance designed to aggregate the data the Steelhead appliance exports. The Cascade Profiler or Cascade Gateway are examples of flow data collectors, which process and present this data in a meaningful way to the administrator. The collector captures
 - Enough information to map the outer-connection to its corresponding inner-connection.
 - The byte and packet reduction for each optimized connection.

- information about which Steelhead appliance interface optimized the connection, including which peer it used during optimization.
- **Analyzer** - A collection of tools used to analyze the data and provide relevant data summaries and graphs. Flow data analyzers are available for free or from commercial sources. An analyzer is often provided in conjunction with a collector.

For smaller networks, the flow data collector and analyzer are typically combined into a single device. For larger networks, a more distributed architecture might be used. In a distributed design, multiple flow data exporters export their data to several flow data collectors which in turn send data back to the flow data analyzer.

Some environments configure NetFlow on the WAN routers to monitor the traffic traversing the WAN. However, when the Steelhead appliances are in place, the WAN routers only see the inner-channel traffic and not the real IP addresses and ports of the client and server. By enabling flow data on the Steelhead appliance, this becomes a non-issue altogether. The Steelhead appliance can export the flow data instead of the router without compromising any functionality. By doing so, the router can spend more CPU cycles on its core functionality: routing and switching of packets.

Before you enable flow data support in your network, consider the following:

- Generating flow data can use large amounts of bandwidth, especially on low bandwidth links and thereby impact Steelhead appliance performance.
- To reduce the amount of data exported, you can export only optimized traffic.

For information about Steelhead appliance MIB and SNMP traps, see the *Steelhead Appliance Management Console User's Guide*.

SNMP Monitoring

This section describes the SNMP traps. It does not list the corresponding clear traps. Every Steelhead appliance supports SNMP traps, and email alerts for conditions that require attention or intervention. An alarm triggers for most (but not every) event and subsequently, the related trap is sent. For most events, when the condition is fixed, the system clears the alarm and sends out a clear trap. The clear traps are useful in determining when an event has been resolved.

RiOS v5.0 supports the following:

- SNMP Version 1
- SNMP Version 2c

RiOS v6.0 or later supports the following:

- SNMP Version 3, which provides authentication through the User-based Security Model (USM).
- View-Based Access Control Mechanism (VACM), which provides richer access control.

RiOS v7.0 or later supports the SNMP v3 authentication with AES 128 and DES encryption described in the following table. Riverbed recommends the following OIDs as a good starting point from which to monitor your deployment. Additional variables can be added or removed as needed.

Note: The following OIDs are for xx55 Steelhead appliances only.

OID	Descriptions
1.3.6.1.4.1.17163.1.1.5.2.1.0	Current total number of optimized connections
1.3.6.1.4.1.17163.1.1.5.2.2.0	Current total number of pass-through connections
1.3.6.1.4.1.17163.1.1.5.2.3.0	Current total number of half-opened (optimized) connections
1.3.6.1.4.1.17163.1.1.5.2.4.0	Current total number of half-closed (optimized) connections
1.3.6.1.4.1.17163.1.1.5.2.5.0	Current number of established (optimized) connections
1.3.6.1.4.1.17163.1.1.5.2.6.0	Current number of active (optimized) connections
1.3.6.1.4.1.17163.1.1.5.2.7.0	Total number of connections
1.3.6.1.4.1.17163.1.1.5.1.1.0	One-minute CPU load in hundredths
1.3.6.1.4.1.17163.1.1.5.1.2.0	Five-minute CPU load in hundredths
1.3.6.1.4.1.17163.1.1.5.1.3.0	Fifteen-minute CPU load in hundredths
1.3.6.1.4.1.17163.1.1.5.1.4.0	Percentage CPU utilization, aggregated across all CPUs, rolling average over the past minute
1.3.6.1.4.1.17163.1.1.5.1.5.1.1.1	A synthetic number numbering the CPUs
1.3.6.1.4.1.17163.1.1.5.1.5.1.2.1	Name of the CPU, also serves as the Index for the table
1.3.6.1.4.1.17163.1.1.5.1.5.1.3.1	Idle time for this CPU
1.3.6.1.4.1.17163.1.1.5.1.5.1.4.1	System time for this CPU
1.3.6.1.4.1.17163.1.1.5.1.5.1.5.1	User time for this CPU
1.3.6.1.4.1.17163.1.1.4.0.8	RAID errors

In multiple CPU systems, the last digit corresponds to the CPU number.

The following table summarizes SNMP traps that represent serious issues and Riverbed recommends that you address them immediately.

Trap and OID	Steelhead Appliance State	Text	Description
procCrash (enterprises.17163.1.1.4.0.1)		A procCrash trap signifies that a process managed by PM has crashed and left a core file. The variable sent with the notification indicates which process crashed.	A process crashed and subsequently restarted by the system. The trap contains the name of the process that crashed. A system snapshot associated with this crash is created on the Steelhead appliance and is accessible through the CLI or the Management Console. Riverbed Support might need this information to determine the cause of the crash. The crashed process automatically restarts and no other action is required on the Steelhead appliance.
procExit (enterprises.17163.1.1.4.0.2)		A procExit trap signifies that a process managed by PM has exited unexpectedly, but not left a core file. The variable sent with the notification indicates which process exited.	A process unexpectedly exited and subsequently restarted by the system. The trap contains the name of the process. The process might have exited automatically due to other process failures on the Steelhead appliance. Review the release notes for known issues related to this process exit. If none exist, Contact Riverbed Support to determine the cause of this event. The crashed process automatically restarts and no other action is required on the Steelhead appliance.
bypassMode (enterprises.17163.1.1.4.0.7)	Critical	The Steelhead appliance has entered bypass (failthru) mode.	The Steelhead appliance entered bypass mode and passes through all traffic unoptimized. This is the result of the optimization service locking up or crashing. It can also happen when the system is first turned on or turned off. If this trap is generated on a system that was previously optimizing and is still running, contact Riverbed Support.
storeCorruption (enterprises.17163.1.1.4.0.9)	Critical	The RiOS data store is corrupted.	Corruption is detected in the RiOS data store. Contact Riverbed Support immediately.
haltError (enterprises.17163.1.1.4.0.12)	Critical	The service is halted due to a software error.	The optimization service halts due to a serious software error. Check to see if a core dump or sysdump was created. If so, retrieve the information and contact Riverbed Support immediately.

Trap and OID	Steelhead Appliance State	Text	Description
serviceError (enterprises.17163.1.1.4.0.13)	Degraded	There has been a service error. Consult the log file.	The optimization service encountered a condition that might degrade optimization performance. Consult the system log for more information.
licenseError (enterprises.17163.1.1.4.0.57)	Critical	The main Steelhead license has expired, been removed, or become invalid.	A license on the Steelhead appliance has been removed, has expired, or is invalid. The alarm clears when a valid license is added or updated.
hardwareError (enterprises.17163.1.1.4.0.58)	Either Critical or Degraded, depending on the state	Hardware error detected.	<p>Indicates that the system has detected a problem with the Steelhead appliance hardware. These issues trigger the hardware error alarm:</p> <ul style="list-style-type: none"> the Steelhead appliance does not have enough disk, memory, CPU cores, or NIC cards to support the current configuration the Steelhead appliance is using a memory Dual In-line Memory Module (DIMM), a hard disk, or a NIC that is not qualified by Riverbed other hardware issues <p>The alarm clears when you add the necessary hardware, remove the unqualified hardware, or resolve other hardware issues.</p>
lanWanLoopError (enterprises.17163.1.1.4.0.63)	Critical	LAN-WAN loop detected. System will not optimize new connections until this error is cleared.	A LAN-WAN network loop has been detected between the LAN and WAN interfaces on a Virtual Steelhead. This can occur when you connect the LAN and WAN virtual NICs to the same vSwitch or physical NIC. This alarm triggers when a Virtual Steelhead starts up, and clears after you connect each LAN and WAN virtual interface to a distinct virtual switch and physical NIC (through the vSphere Networking tab) and then reboot the Virtual Steelhead.

Trap and OID	Steelhead Appliance State	Text	Description
optimizationServiceStatusError (enterprises.17163.1.1.4.0.64)	Critical	Optimization service currently not optimizing any connections.	<p>The optimization service has encountered an optimization service condition. The message indicates the reason for the condition:</p> <ul style="list-style-type: none"> • optimization service is not running This message appears after a configuration file error. For more information, review the Steelhead appliance logs. • in-path optimization is not enabled This message appears if an in-path setting is disabled for an in-path Steelhead appliance. For more information, review the Steelhead appliance logs. • optimization service is initializing This message appears after a reboot. The alarm clears on its own; no other action is necessary. For more information, review the Steelhead appliance logs. • optimization service is not optimizing This message appears after a system crash. For more information, review the Steelhead appliance logs. • optimization service is disabled by user This message appears after entering the CLI command no service enable or shutting down the optimization service from the Management Console. For more information, review the Steelhead appliance logs. • optimization service is restarted by user This message appears after the optimization service is restarted from either the CLI or Management Console. You might want to review the Steelhead appliance logs for more information.

Trap and OID	Steelhead Appliance State	Text	Description
storageProfSwitchFailed (enterprises.17163.1.1.4.0.73)	Either Critical or Needs Attention, depending on the state	Storage profile switch failed	<p>An error has occurred while repartitioning the disk drives during a storage profile switch. A profile switch changes the disk space allocation on the drives, clears the Granite and VSP data stores, and repartitions the data stores to the appropriate sizes.</p> <p>You switch a storage profile by entering the disk-config layout CLI command at the system prompt or by choosing Configure > System Settings > Disk Management on an EX or EX+Granite Steelhead appliance and selecting a storage profile.</p> <p>These reasons can cause a profile switch to fail:</p> <ul style="list-style-type: none"> • RiOS cannot validate the profile. • The profile contains an invalid upgrade or downgrade. • RiOS cannot clean up the existing VDMKs. During clean up RiOS uninstalls all slots and deletes all backups and packages. <p>When you encounter this error, try to switch the storage profile again. If the switch succeeds, the error clears. If it fails, RiOS reverts the Steelhead appliance to the previous storage profile.</p> <ul style="list-style-type: none"> • If RiOS is unable to revert the Steelhead appliance to the previous storage profile, the alarm status becomes critical. • If RiOS successfully reverts the Steelhead appliance to the previous storage profile, the alarm status displays needs attention.

Trap and OID	Steelhead Appliance State	Text	Description
flashProtectionFailed (enterprises.17163.1.1.4.0.75)	Critical	Flash disk hasn't been backed up due to not enough free space on /var filesystem.	Indicates that the USB flash drive has not been backed up because there is not enough available space in the /var filesystem directory. Examine the /var directory to see if it is storing an excessive amount of snapshots, system dumps, or TCP dumps that you could delete. You could also delete any RiOS images that you no longer use.
datastoreNeedClean (enterprises.17163.1.1.4.0.76)	Critical	The data store needs to be cleaned.	You need to clear the RiOS data store. To clear the data store, choose Configure > Maintenance > Services and select the Clear Data Store check box before restarting the appliance. Clearing the data store degrades performance until the system repopulates the data.

If an error condition exists, there are several alarms that are generated along with the SNMP traps. If the email feature is configured, you receive an email notification in addition to the alarms.

To limit the number of alarms generated over a given period of time, use the **stats alarm <alarm name> rate-limit count <thresholds> <count>** command.

There are three sets of thresholds—short, medium and long. Each has a window, which is several seconds, and a maximum count. If, for any threshold, the number of alarms exceeds the maximum during the window, an alarm is not generated and emails are not sent.

For more details on configuring SNMP and other important traps, see the *Steelhead Appliance Management Console User's Guide*.

Configuring SNMP v3 Authentication and Privacy

RiOS v7.0 or later includes privacy to the SNMP v3 feature to support authentication and privacy encryption of SNMPv3 messages. You can use AES 128 and DES to send an SNMP v3 encryption for GET action.

All SNMPv3 passwords (authentication/privacy) are stored as hashed (MD5/SHA), and they are all master keys, even if you provide plain text password during configuration.

An SNMP agent runs in every Steelhead appliance that supports SNMP GET request action. Among the techniques to secure SNMP traffic, such as access control lists, you can use SNMP v3 to provide authentication and privacy. The main benefit for SNMP v3 authentication is to ensure the integrity of SNMP traffic, while privacy provides encryption protecting data from being seen by a third party.

Configuring an SNMP v3 GET request encryption is a two-part process:

- Configure USM user

The user corresponds with the authentication and privacy mechanism that a management station uses to access the Steelhead appliance.

■ Configure ACLs

To configure the ACLs, you need to add or edit a group, view and access policy. You cannot add an access policy with a group and a view. Security names are not supported by SNMPv3. To restrict SNMP v3 USM users from polling a specific subnet, use the RiOS ACL feature on the Configure > Security > Management ACL page.

Views represent the OIDs a management station is allowed to access. You can create multiple views and restrict specific OIDs. A view starts with the highest level OID that you specify, and you can view all OIDs further down in the hierarchy, unless you specifically restrict them. You can only view OIDs in the hierarchy.

You must associate a group with a view. After you associate a group with a view, you can define an access policy to link the user, group, and view together.

In the following procedure show an example of user *Cascade* created with SHA authentication and AES encryption for privacy.

To configure an USM user

1. From the Steelhead Management Console, choose Configure > System Settings > SNMP v3.
2. Select the Add a New User tab.

Figure 20-3. Add a New USM User

The screenshot shows the 'Configure > System Settings > SNMP v3' page. Under the 'Users:' section, the 'Add a New User' tab is selected. The form includes the following fields and options:

- User Name:** Cascade
- Authentication Protocol:** SHA
- Authentication:** Supply a Password
- Password:** (masked with dots, note: at least 8 characters)
- Password Confirm:** (masked with dots)
- ☒ **Use Privacy Option**
- Privacy Protocol:** AES
- Privacy:** Supply a Password
- Privacy Password:** (masked with dots, note: at least 8 characters)
- Privacy Password Confirm:** (masked with dots)
- Add** button

3. Select Use Privacy Option.
4. Select AES or DES from the Privacy Protocol drop-down list.
5. Select any of the options in the Privacy drop-down list and complete any corresponding steps.

[Figure 20-3](#) shows Supply a Password and the corresponding password.

6. Click **Add**.

In the following procedure show an example of group *Profiler* created, and then user Cascade is associated with the group Profiler.

To configure SNMP ACLs

1. From the Steelhead Management Console, choose **Configure > System Settings > SNMP ACLs**.
2. Select the **Add a New Group** tab.

Figure 20-4. Add a New Group

A group is one or more entries of the form security-model:security-name.

Groups:

▼ Add a New Group — Remove Selected

Specify the group name and select the security models. For v1 and v2c security models, select the security name. For usm security models, select the user name.

Group Name:

Security Model and Name Pairs:

Group Name	Security Models, Names
No Groups.	

3. Specify a group name.
4. Select usm and select the user you created in [“To configure an USM user”](#).
5. Click **Add**.

6. Select the Add a New View tab.

Figure 20-5. Add a New View

Add OIDs that should be included or excluded from this view

Views:

▼ Add a New View — Remove Selected

View Name:

Includes:

(one .x.y.z per line)

Excludes:

(one .x.y.z per line)

View Name	Includes	Excludes
No Views.		

7. Specify a view name.
8. Specify the OIDs to include and exclude from the view.
9. Click Add.
10. Select the Add a New Policy tab.

Figure 20-6. Add a New Policy

Access Policies are rules for the agent to decide how to process a request.

Access Policies:

▼ Add a New Access Policy — Remove Selected

Group Name:

Security Level:

Read View:

Group Name	Security Level	Read View
No Access Policies.		

11. Select the group name you created from the Group Name drop-down list.
12. Select AuthPriv from the Security Level drop-down list.

13. Select the view you created from the Read View drop-down list.

14. Click **Add**.

You can verify your configuration in Wireshark. Make sure the SNMP PDUs are encrypted.

Figure 20-7. Wireshark Verification

No.	Time	Delta Time	Conv	VLAN	Src	Dest	S Port	D Port	Proto	Len	DSCP	Info
3874	391.899305				192.168.0.17	192.168.0.10	58700	161	SNMP	240	0	encryptedPDU: privkey unknown
3875	391.899881				192.168.0.10	192.168.0.17	161	58700	SNMP	240	0	encryptedPDU: privkey unknown
3896	396.898033				192.168.0.17	192.168.0.10	58701	161	SNMP	223	0	encryptedPDU: privkey unknown
3897	396.898695				192.168.0.10	192.168.0.17	161	58701	SNMP	240	0	encryptedPDU: privkey unknown
3908	401.897648				192.168.0.17	192.168.0.10	58702	161	SNMP	223	0	encryptedPDU: privkey unknown

To decrypt the SNMP packets for further troubleshooting

1. From the Wireshark menu, choose Edit > Preferences > Protocols > SNMP.
2. Select the Edit for the SNMP Users window.

Figure 20-8. SNMP Users Window

SNMP Users: New - Profile: Def...

Engine ID:

Username:

Authentication model: MD5

Password:

Privacy protocol: DES

Privacy password:

OK Cancel

3. Complete the information in the SNMP Users window.

Engine ID is available on the Steelhead appliance through the **show snmp CLI** command or near the end of the running-configuration. The username, authentication model, password, privacy protocol and privacy password are the same settings you configured for the SNMPv3 user on the Steelhead appliance.

4. Click **OK**.

Wireshark decrypts the SNMP encrypted packets and you can analyze further for troubleshooting.

CHAPTER 21 Troubleshooting Steelhead Appliance Deployment Problems

This chapter describes common deployment problems and solutions. This chapter includes the following sections:

- [“Common Deployment Issues” on page 431](#)
- [“MTU Sizing” on page 445](#)

For details on Steelhead appliance installation issues, see the *Steelhead Appliance Installation and Configuration Guide*.

For details on the factors to consider before you deploy the Steelhead appliance, see [“Choosing the Right Steelhead Appliance” on page 16](#).

Common Deployment Issues

This section contains solutions to the following deployment issues:

- [“Duplex Mismatches” on page 432](#)
- [“Duplex Mismatches” on page 432](#)
- [“Inability to Access Files During a WAN Disruption” on page 434](#)
- [“Network Asymmetry” on page 434](#)
- [“Unknown \(or Unwanted\) Steelhead Appliance Appears on the Current Connections List” on page 436](#)
- [“Outdated Antivirus Software” on page 437](#)
- [“Router CPU Spikes After WCCP Configuration” on page 438](#)
- [“Packet Ricochets” on page 437](#)
- [“Router CPU Spikes After WCCP Configuration” on page 438](#)
- [“Server Message Block Signed Sessions” on page 439](#)
- [“Unavailable Opportunistic Locks” on page 444](#)
- [“Underutilized Fat Pipes” on page 445](#)

Duplex Mismatches

This section describes common problems that can occur in networks in which duplex settings do not match. Duplex mismatch occurs when the speed of a network interface that is connected to the Steelhead appliance does not match.

The number one cause of poor performance issues with Steelhead appliance installations is duplex mismatch. A duplex mismatch can cause performance degradation and packet loss.

Signs of duplex mismatch:

- You cannot connect to an attached device.
- You can connect with a device when you choose auto-negotiation, but you cannot connect with the same device when you manually set the speed or duplex.
- Little or no performance gains.
- Loss of network connectivity.
- Intermittent application or file errors.
- All of your applications are slower after you have installed in-path Steelhead appliances.

To determine whether the slowness is caused by a duplex mismatch

1. Create a pass-through rule for the application on the client-side Steelhead appliance and ensure that the rule is at the top of the in-path rules list. You add a pass-through rule with the command **in-path rule pass-through** or you can use the Management Console.
2. Restart the application.
3. Check that all connections related to the application are being passed through. If all connections related to the application are being passed through and the performance of the application does not return to the original levels, the slowness is most likely due to duplex mismatch.

The following sections describe several possible solutions to duplex mismatch.

Solution: Manually Set Matching Speed and Duplex

One solution for mismatched speed and duplex settings is to manually configure the settings.

1. Manually set (that is, hard set) matching speed and the duplex settings for the following four ports:
 - Devices (switches) connected on the Steelhead appliance LAN port
 - Devices (routers) connected on the Steelhead appliance WAN port
 - The Steelhead appliance LAN port
 - The Steelhead appliance WAN port

Riverbed recommends the following speeds:

- Fast Ethernet Interfaces: 100 megabits full duplex
- Gigabit Interfaces: 1000 megabits full duplex

For more details, see the Riverbed Knowledge Base article, *Problems manually setting 1000 Mbps/Full on Steelhead*, at <https://supportkb.riverbed.com/support/index?page=content&id=s14623>.

Riverbed recommends that you avoid using half-duplex mode whenever possible. If you are using a modern interface, and it appears to not support full duplex, double check the duplex setting. It is likely that one side is set to auto and the other is set to fixed. To manually change interface speed and duplex settings, use the command **interface**. For details, see the *Riverbed Command-Line Interface Reference Manual*

2. Verify that each of the above devices:

- have settings that match in optimizing mode.
- is configured to see interface speed and duplex settings, using the command **show configuration**. By default, the Steelhead appliance automatically negotiates speed and duplex mode for all data rates and supports full duplex mode and flow control. To change interface speed and duplex settings, use the command **interface**.
- have settings that match in bypass mode.
- are not showing any errors or collisions.
- does not have a half-duplex configuration (forced or negotiated) on either the WAN or the LAN.
- has at least 100 Mbps speed, forced or negotiated, on the LAN.
- has network connectivity in optimization and in failure mode.

For details on failure mode, see [“Failure Modes” on page 174](#).

3. Test connectivity with the Steelhead appliance powered off. This ensures that the Steelhead appliance does not sever the network in the event of a hardware or software problem. This must be done last, especially after making any duplex changes on the connected devices.
4. If the Steelhead appliance is powered off and you cannot pass traffic through it, verify that you are using the correct cables for all devices connected to the Steelhead appliance. The type of cable is determined by the device connecting to the Steelhead appliance:
- Router to Steelhead appliance: use a crossover cable.
 - Switch to Steelhead appliance: use a straight-through cable.
 - Do not rely on Auto MDI/MDI-X to determine which cables you are using.

For details on cables, see [“Choosing the Correct Cables” on page 177](#).

5. Use a cable tester to verify that the Steelhead appliance in-path interface is functioning properly: turn off the Steelhead appliance, and connect the cable tester to the LAN and WAN port. The test result must show a crossover connection.
6. Use a cable tester to verify that all of the cables connected to the Steelhead appliance are functioning properly.

For details on how to choose the correct cables, see [“Choosing the Correct Cables” on page 177](#).

Solution: Use an Intermediary Switch

If you have tried to manually set matching speed and duplex settings, and duplex mismatch still causes slow performance and lost packets after you deploy in-path Steelhead appliances, introduce an intermediary switch that is more compatible with both existing network interfaces. Riverbed recommends that you use this option only as a last option.

Important: To use an intermediary switch, you must also change your network cables appropriately.

Inability to Access Files During a WAN Disruption

If your network requires that clients have continuous access to files, even in the event of network disruptions that prevent access over the WAN to the origin server on which the files are located, consider using PFS.

PFS is an optional integrated virtual file server that allows you to store copies of files on the Steelhead appliance with Windows file access, creating several options for transmitting data between remote offices and centralized locations with improved performance and functions. Data is configured into file shares by PFS, and the shares are periodically synchronized transparently in the background, over the optimized connection of the Steelhead appliance. PFS leverages the integrated disk capacity of the Steelhead appliance to store file-based data in a format that allows it to be retrieved by NAS clients.

For details on PFS, see [“Proxy File Services Deployments” on page 381](#)

Solution: Use Proxy File Service

If you are using Steelhead appliance Models 520, 1010, 1020, 1520, 2020, 3010, 3020, 3520, 5010, or 6120, you can configure PFS to ensure that remote sites can access files even when a WAN disruption prevents access to the origin server on which files are located.

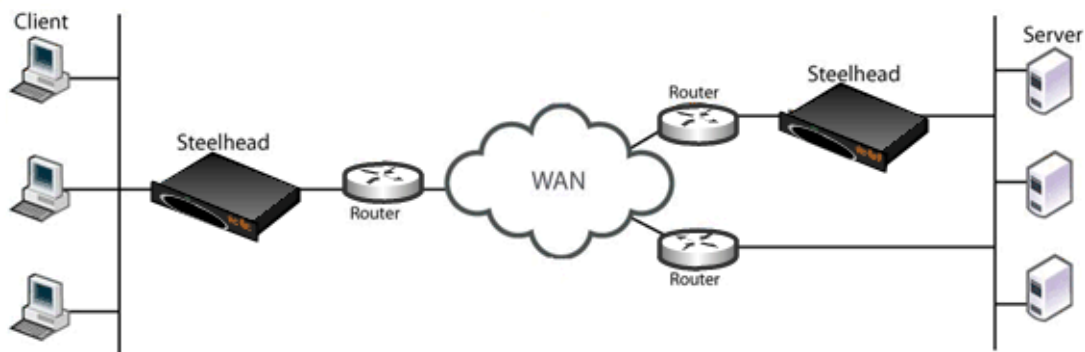
Network Asymmetry

If some of the connections in a network are optimized and some are passed through unoptimized, it might be due to network asymmetry. Network asymmetry causes a client request to traverse a different network path than the server response. Network asymmetry can also break connections.

If SYN packets that traverse from one side of the network are optimized, but SYN packets that traverse from the opposite side of the network are passed-through unoptimized, it is a symptom of network asymmetry.

[Figure 21-1](#) shows an asymmetric server-side network in which a server response can traverse a path (the bottom path) in which a Steelhead appliance is not installed.

Figure 21-1. Server-Side Asymmetric Network



The following sections describe several possible solutions to network asymmetry.

With RiOS v3.0.x or later, you can configure your Steelhead appliances to automatically detect and report asymmetric routes within your network. Whether asymmetric routing is automatically detected by Steelhead appliances or is detected in some other way, use the solutions described in the following sections to work around it.

For details on configuring auto-detection of asymmetric routes, see the *Steelhead Appliance Management Console User's Guide*.

Solution: Use Connection Forwarding

For a network connection to be optimized, packets traveling in both network directions (from server to client and from client to server) must pass through the same client-side and server-side Steelhead appliance. In networks in which asymmetric routing occurs because client requests or server responses can traverse different paths, you can solve it by:

- ensuring that there is a Steelhead appliance installed on every possible path a packet can traverse. You would install a second server-side Steelhead appliance, covering the bottom path. For details, see [Figure 21-1](#).
- setting up connection forwarding to route packets that traversed one Steelhead appliance in one direction to traverse the same Steelhead appliance in the opposite direction. Connection forwarding can be configured on the client-side or server-side of a network.

To set up connection forwarding, use the Management Console or CLI as described in the *Steelhead Appliance Management Console User's Guide* and the *Riverbed Command-Line Interface Reference Manual*.

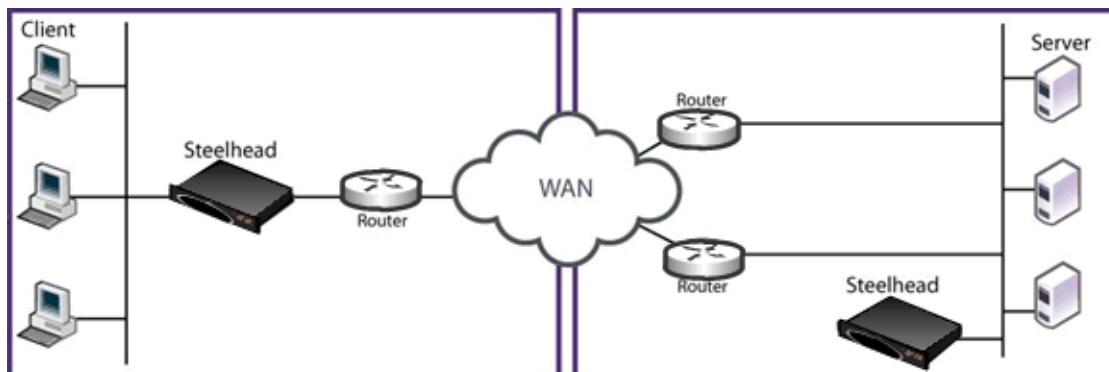
For more information, see [“Connection Forwarding” on page 42](#).

Solution: Use Virtual In-Path Deployment

Because a connection cannot be optimized unless packets traveling in both network directions pass through the same client-side Steelhead appliance and the same server-side Steelhead appliance, you can use a virtual in-path deployment to solve network asymmetry.

In the example network shown in [Figure 21-1](#), changing the server-side Steelhead appliance that is deployed in-path on the top server-side path to a virtual in-path deployment, ensures that all server-side traffic passes through the server-side Steelhead appliance.

Figure 21-2. Virtual In-Path Deployment to Solve Network Asymmetry



A virtual in-path deployment differs from a physical in-path deployment in that a packet redirection mechanism directs packets to Steelhead appliances that are not in the physical path of the client or server. Redirection mechanisms include a Layer-4 switch (or server load balancer), WCCP, and PBR. These redirection mechanisms are described in:

- [“Virtual In-Path Deployments” on page 217](#)
- [“Out-of-Path Deployments” on page 335](#)
- [“WCCP Virtual In-Path Deployments” on page 221](#)
- [“Policy-Based Routing Virtual In-Path Deployments” on page 255](#)

Solution: Deploy a Four-Port Steelhead Appliance

If you have a Steelhead appliance that supports a Four-Port Copper Gigabit-Ethernet PCI-X card, you can deploy it to solve network asymmetry in which a two-port Steelhead appliance or one of the solutions described in previous sections is not successful.

For example, instead of the two-port Steelhead appliance deployed to one server-side path as shown [Figure 21-1](#), you deploy a four-port Steelhead appliance on the server-side of the network. All server-side traffic passes through the four-port Steelhead appliance and asymmetric routing is eliminated.

For details on two- and four-port Steelhead appliances, see the *Network Interface Card Installation Guide*.

Unknown (or Unwanted) Steelhead Appliance Appears on the Current Connections List

Enhanced auto-discovery greatly reduces the complexities and time it takes to deploy Steelhead appliances. It works so seamlessly that occasionally it has the undesirable effect of peering with Steelhead appliances on the Internet that are not in your organization's management domain or your corporate business unit. When an unknown (or unwanted) Steelhead appliance appears connected to your network, you can create a peering rule to prevent it from peering and remove it from your list of peers. The peering rule defines what to do when a Steelhead appliance receives an auto-discovery probe from the unknown Steelhead appliance.

To prevent an unknown Steelhead appliance from peering

1. Choose **Configure > Optimization > Peering Rules**.
2. Click **Add a New Peering Rule**.
3. Select **Passthrough** as the rule type.
4. Specify the source and destination subnets. The source subnet is the remote location network subnet (in the format XXX.XXX.XXX.XXX/XX). The destination subnet is your local network subnet (in the format XXX.XXX.XXX.XXX/XX).
5. Click **Add**.

In this example, the peering rule passes through traffic from the unknown Steelhead appliance in the remote location.

When you use this method and add a new remote location in the future, you must create a new peering rule that accepts traffic from the remote location. Place this new **Accept** rule before the **Pass-through** rule.

If you do not know the network subnet for the remote location, there is another option: you can create a peering rule that allows peering from your corporate network subnet and denies it otherwise. For example, create a peering rule that accepts peering from your corporate network subnet and place it as the first rule in the list.

Next, create a second peering rule to pass through all other traffic. In this example, when the local Steelhead appliance receives an auto-discovery probe, it checks the peering rules first (from top to bottom). If it matches the first Accept rule, the local Steelhead appliance peers with the other Steelhead appliance. If it does not match the first Accept rule, the local Steelhead appliance checks the next peering rule, which is the pass-through rule for all other traffic. In this case, the local Steelhead appliance just passes through the traffic, and does not peer with the other Steelhead appliance.

After you add the peering rule, the unknown Steelhead appliance appears in the Current Connections report as a Connected Appliance until the connection times out. After the connection becomes inactive, it appears dimmed. To remove the unknown appliance completely, restart the optimization service.

Outdated Antivirus Software

After installing Steelhead appliances, if application access over the network does not speed up or certain operations on files (such as dragging and dropping) speed up greatly but application access does not, it might be due to old antivirus software installed on a network client.

For similar problems, see:

- [“Server Message Block Signed Sessions” on page 439](#)
- [“Unavailable Opportunistic Locks” on page 444](#)

Solution: Upgrade Antivirus Software

If it is safe to do so, temporarily disable the antivirus software and try opening files. If performance improves with antivirus software disabled, Riverbed recommends that you upgrade the antivirus software.

If performance does not improve with antivirus software disabled or after upgrading antivirus software, contact Riverbed Support site at <https://support.riverbed.com>.

Packet Ricochets

Signs of packet ricochet are:

- Network connections fail on their first attempt but succeed on subsequent attempts.
- The Steelhead appliance on one or both sides of a network has an in-path interface that is different from that of the local host.
- You have not defined any in-path routes in your network.
- Connections between the Steelhead appliance and the clients or server are routed through the WAN interface to a WAN gateway, and then they are routed through a Steelhead appliance to the next-hop LAN gateway.
- The WAN router drops SYN packets from the Steelhead appliance before it issues an ICMP redirect.

Solution: Add In-Path Routes

To prevent packet ricochet, add in-path routes to local destinations. For details, see *Steelhead Appliance Management Console User's Guide*.

For details on packet ricochet, see [“In-Path Redundancy and Clustering Examples” on page 184](#).

Solution: Use Simplified Routing

You can also use simplified routing to prevent packet ricochet. To configure simplified routing, use the CLI command **in-path simplified routing** or the Management Console.

For details on simplified routing and how to configure it, see the *Riverbed Command-Line Interface Reference Manual* or the *Steelhead Appliance Management Console User's Guide*.

Router CPU Spikes After WCCP Configuration

If the CPU usage of the router spikes after WCCP configuration, it might be because you are not using a WCCP-compatible Cisco IOS release, or because you must use inbound redirection.

The following sections describe several possible solutions to router CPU spike after WCCP configuration.

Solution: Use Mask Assignment instead of Hash Assignment

The major difference between the hash and mask assignment methods lies in the way traffic is processed within the router/switch. With a mask assignment, traffic is processed entirely in the hardware, which means the CPU of the switch is minimal. A hash assignment uses the switch CPU for part of the load distribution calculation and hence places a significant load on the switch CPU. The mask assignment method was specifically designed for hardware-based switches and routers (such as Cisco 3560, 3750, 4500, 6500, and 7600).

For details on mask assignment, see [“WCCP Virtual In-Path Deployments” on page 221](#).

Solution: Check Internetwork Operating System Compatibility

Because WCCP is not fully integrated in every IOS release and on every platform, ensure that you are running a WCCP-compatible IOS release. If you have questions about the WCCP compatibility of your IOS release, contact Riverbed Support site at <https://support.riverbed.com>.

If you are certain that you are running a WCCP-compatible IOS release and you experience router CPU spike after WCCP configuration, review the remaining sections for possible solutions.

Solution: Use Inbound Redirection

One possible solution to router CPU spike after WCCP configuration is to use inbound redirection instead of outbound redirection. Inbound redirection ensures that the router does not waste CPU cycles consulting the routing table before handling the traffic for WCCP redirection.

For details on redirection, see [“WCCP Virtual In-Path Deployments” on page 221](#)

Solution: Use Inbound Redirection with Fixed-Target Rules

If inbound redirection, as described in [“Solution: Use Inbound Redirection” on page 438](#), does not solve router CPU spike after WCCP is configured, try using inbound redirection with a fixed-target rule between Steelhead appliances. The fixed-target rule can eliminate one redirection interface.

Fixed-target rules directly specify server-side Steelhead appliances near the target server that you want to optimize. You determine which servers you would like the Steelhead appliance to optimize (and, optionally, which ports), and add fixed-target rules to specify the network of servers, ports, and out-of-path Steelhead appliances to use.

For details on how to configure inbound redirection and fixed-target rules, see [“WCCP Virtual In-Path Deployments” on page 221](#)

Solution: Use Inbound Redirection with Fixed-Target Rules and Redirect List

If the solutions described in the previous sections do not solve router CPU spike after WCCP is configured, try using inbound redirection with a fixed-target rule and a redirect list. A redirect list can reduce the load on the router by limiting the amount of unnecessary traffic that is redirected by the router.

For details, see [“WCCP Virtual In-Path Deployments” on page 221](#)

Solution: Base Redirection on Ports Rather than ACLs

If the solutions described in the previous sections do not solve router CPU spike after WCCP configuration, consider basing traffic redirection on specific port numbers rather than using ACLs.

Solution: Use PBR

If the solutions described in the previous sections do not solve router CPU spike after WCCP configuration, consider using PBR instead of WCCP.

For details on PBR, see [“Policy-Based Routing Virtual In-Path Deployments” on page 255](#)

Server Message Block Signed Sessions

This section provides a brief overview of problems that can occur with Windows Server Message Block (SMB) signing. For details on SMB signing, the performance cost associated with it, and solutions to it, see the *Steelhead Appliance Management Console User's Guide*.

If network connections appear to be optimized but there is no performance difference between a cold and warm transfer, it might be due to SMB-signed sessions.

SMB-signed sessions support compression and RiOS SDR, but render latency optimization (for example read-ahead, and write-behind) unavailable.

Signs of SMB signing:

- Access to some Windows file servers across a WAN is slower than access to other Windows file servers across the WAN.
- Connections are shown as optimized in the Management Console.
- The results of a TCP dump show low WAN utilization for files where their contents do not match existing segments in the segment store.
- Copying files via FTP from the slow server is much faster than copying the same files via mapped network drives (CIFS).

When copying FTP from a slow server is much faster than copying from the same server via a mapped network drive, the possibility of other network problems (such as duplex mismatch or network congestion) with the server is ruled out.

- Log messages in the Management Console such as:

```
error=SMB_SHUTDOWN_ERR_SEC_SIG_ENABLED
```

The following sections describe possible solutions to SMB-signed sessions.

For similar problems, see:

- [“Unknown \(or Unwanted\) Steelhead Appliance Appears on the Current Connections List” on page 436](#)
- [“Unavailable Opportunistic Locks” on page 444](#)

Solution: Fully Optimize SMB-Signed Traffic

Before you use any of the following solutions, configure your Steelhead appliance to optimize SMB-signed traffic. For details on how to configure SMB-signed traffic, see the *Steelhead Appliance Deployment Guide - Protocols*.

If you do not have the privileges or the correct information for SMB-signed traffic optimization, try the following solutions.

Solution: Enable Secure-CIFS

Enable Secure-CIFS using the CLI command **protocol cifs secure-sig-opt enable**.

The Secure-CIFS feature automatically stops Windows SMB signing. SMB signing prevents the Steelhead appliance from applying full optimization on CIFS connections and significantly reduces the performance gain from a Steelhead appliance deployment. SMB-signed sessions support compression and RiOS SDR, but render latency optimization (read-ahead, write-behind) unavailable.

With Secure-CIFS enabled, you must consider the following factors:

- If the client-side machine has Required signing, enabling the Secure-CIFS feature prevents the client from connecting to the server.
- If the server-side machine has Required signing, the client and the server connect but you cannot perform full latency optimization with the Steelhead appliance. (Domain Controllers default to Required.)

For details on SMB signing, see the *Steelhead Appliance Installation and Configuration Guide*.

Alternatively, if your deployment requires SMB signing, you can optimize signed CIFS messages by selecting Enable SMB Signing in the Optimization > CIFS page of the Management Console. Before you enable SMB signing, make sure you *disable* Optimize Connections with Security Signatures. For detailed information about optimizing signed CIFS messages, including procedures for your Windows server, see the *Steelhead Appliance Management Console User's Guide*.

Note: Secure-CIFS is enabled by default beginning with RiOS v2.x.

Tip: If a log file shows messages such as error=SMB_SHUTDOWN_ERR_SEC_SIG_REQUIRED, use the solution described in [“Solution: Disable SMB Signing with Active Directory” on page 441](#). Enabling secure-CIFS has no effect when SMB signing has been set to required.

For details, see the *Steelhead Management Console Online Help* or the *Steelhead Appliance Management Console User's Guide*.

Solution: Disable SMB Signing with Active Directory

If you have tried enabling Secure-CIFS as described in [“Solution: Enable Secure-CIFS” on page 440](#) but SMB signing still occurs, consider using Active Directory (AD) to disable SMB signing requirements on servers or clients.

If the Security Signature feature does not disable SMB signing, you must revise the default SMB registry parameters. SMB signing is controlled by the following registry parameters:

```
enablesecuritysignature (SSEn)  
requiresecuritysignature (SSReq)
```

The registry settings are located in:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\lanmanserver\parameters
```

The following table summarizes the default SMB signing registry parameters.

Machine Role	SSEn	SSReq
Client/Workstation	ON	OFF
Member Server	OFF	OFF
Domain Controller	ON	ON

With these default registry parameters, SMB signing is negotiated in the following manner:

- SMB/CIFS exchanges between the Client/Workstation and the Member Server are not signed.
- SMB/CIFS exchanges between the Client/Workstation and the Domain Controller are always signed.

The following table lists the complete matrix for SMB registry parameters that ensure full optimization (that is, bandwidth and latency optimization) using the Steelhead appliance.

Number	Parameters on Workstation		Parameters on Server		Result
	SSReq	SSEn	SSReq	SSEn	
1	OFF	OFF	OFF	OFF	Signature Disabled: Steelhead full optimization
2	OFF	OFF	OFF	ON	Signature Disabled: Steelhead full optimization
3	OFF	OFF	ON	ON	Cannot establish session
4*	OFF	OFF	ON	ON	Cannot establish session
5	OFF	ON	OFF	OFF	Signature Disabled: Steelhead full optimization
6	OFF	ON	OFF	ON	Signature Enabled; Steelhead bandwidth optimization
7	OFF	ON	ON	ON	Signature Enabled; Steelhead bandwidth optimization
8*	OFF	ON	OFF	ON	Signature Enabled; Steelhead bandwidth optimization
9	ON	ON	OFF	OFF	Cannot establish session
10*	ON	ON	OFF	ON	Signature Enabled; Steelhead bandwidth optimization
11	ON	ON	ON	ON	Signature Enabled; Steelhead bandwidth optimization
12	ON	ON	OFF	ON	Signature Enabled; Steelhead bandwidth optimization
13+	ON	OFF	OFF	OFF	Cannot establish session
14+	ON	OFF	OFF	ON	Signature Enabled; Steelhead bandwidth optimization
15+	ON	OFF	ON	ON	Signature Enabled; Steelhead bandwidth optimization
16+	ON	OFF	OFF	ON	Signature Enabled; Steelhead bandwidth optimization

Note: Rows with an asterisk (*) and a plus sign (+) are illegal combinations of SSReq and SSEn on the server and the workstation respectively.

This table represents behavior for Windows 2000 workstations and servers with service pack 3 and Critical Fix Q329170. Prior to the critical fix, the security signature feature was not enabled or enforced even on domain controllers.

Each computer has the following set of parameters: one set for the computer as a server and the other set for the computer as a client.

Note: On the client, if SMB signing is set to Required, do not disable it on the server. For the best performance, enable the clients, disable the file servers, and enable domain controllers.

The following procedures assume that you have installed and configured the Steelhead appliances in your network.

To disable SMB signing on Windows 2000 Domain Controllers, member servers, and clients

1. Open Active Directory Users and Computers on the Domain Controller.
2. Right-click Domain Controllers and select Properties.
3. Select the Group Policy tab.
4. Select Default Domain Controllers Policy and click **Edit**.
5. Select Default Domain Controllers Policy / Computer Configuration / Windows Settings / Security Settings / Local Policies / Security Options.
6. Disable Digitally sign client communication (always) and Digitally sign server communication (always).
7. Disable Digitally sign client communication (when possible) and Digitally sign server communication (when possible).
8. Reboot all the Domain Controllers and member servers that you want to optimize.

Tip: You can also open a command prompt and enter `gpupdate.exe /Force` that forces the group policy you just modified to become active without rebooting.

You can verify that SMB signing has been disabled on your domain controllers, member servers, and clients. The following procedures assume that you have installed and configured the Steelhead appliances in your network.

To verify that SMB signing has been disabled

1. Copy some files in Windows from the server to the client through the Steelhead appliances.
2. Connect to the Management Console. For detailed information, see the *Steelhead Appliance Management Console User's Guide*.
3. On the server-side Steelhead appliance choose, Reports > Diagnostics > System Logs.
4. Look for the SMB signing warnings (in red). For example, look for the following text:
`SFE: error=SMB_SHUTDOWN_ERR_SEC_SIG_ENABLED`
5. If you see error messages, repeat [Step 6](#) and [Step 7](#) in procedure [“To disable SMB signing on Windows 2000 Domain Controllers, member servers, and clients”](#).

To disable SMB signing on Windows 2003 Domain Controllers, member servers, and clients

1. Open Active Directory Users and Computers on the Domain Controller.
2. Right-click Domain Controllers and select Properties.
3. Select the Group Policy tab.
4. Click Default Domain Controllers Policy.

5. Click **Edit**.
6. Click Default Domain Controllers Policy/Computer Configuration/Windows Settings/Security Settings/Local Policies/Security Options.
7. Reboot all the Domain Controllers and member servers that you want to optimize.

Unavailable Opportunistic Locks

If a file is not optimized for more than one user at a time, it might be because an application lock on it prevents other applications and the Steelhead appliance from obtaining exclusive access to it. Without an exclusive lock, the Steelhead appliance cannot perform latency (for example, read-ahead and write-behind) optimization on the file.

Without opportunistic locks (oplocks), RiOS SDR and compression are performed on file contents, but the Steelhead appliance cannot perform latency optimization because data integrity cannot be ensured without exclusive access to file data.

The following are signs of unavailable oplocks:

- Within a WAN:
 - A client, PC1, in a remote office across the WAN can open a file it previously opened in just a few seconds.
 - Another client, PC2, on the WAN has also previously opened the file but cannot open it quickly because PC1 has it open. While PC1 has the file open, it takes PC2 significantly longer to open the file.
 - When PC1 closes the file, PC2 can once again open it quickly. However, because PC2 has the file open, PC1 cannot open it quickly; it takes significantly longer for PC1 to open the file because PC2 has it open.
 - If no client has the file open and PC1, PC2, and a third client on the WAN (PC3) simultaneously copy but do not open the file, each client can copy the file quickly and in nearly the same length of time.
- The results of a tcpdump show that WAN utilization is low for files that take a long time to open.
- In the Management Console, slow connections appear optimized.

Tip: You can check connection bandwidth reduction in the Bandwidth Reduction report in the Management Console.

For similar problems, see:

- [“Unknown \(or Unwanted\) Steelhead Appliance Appears on the Current Connections List” on page 436](#)
- [“Server Message Block Signed Sessions” on page 439](#)

Solution: None Needed

To prevent any compromise to data integrity, the Steelhead appliance only accelerates access to data when exclusive access is available. When unavailable oplocks prevent the Steelhead appliance from performing latency optimization, the Steelhead appliance still performs RiOS SDR and compression on the data. Therefore, even without the benefits of latency optimization, Steelhead appliances might still increase WAN performance, but not as effectively as when application optimized connections are available.

Underutilized Fat Pipes

A *fat pipe* is a network that can carry large amounts of data without significantly degrading transmission speed. If you have a fat pipe that is not being fully utilized and you are experiencing WAN congestion, latency, and packet loss as a result of the limitations of regular TCP, consider the solutions outlined in this section.

Solution: Enable High-Speed TCP

To better utilize fat pipes such as in GigE WANs, consider enabling High Speed TCP (HS-TCP). HS-TCP is a feature that you can enable on Steelhead appliances to ease WAN congestion caused by limitations with regular TCP that results in packet loss. Enabling the HS-TCP feature enables more complete utilization of *long fat pipes* (high-bandwidth, high-delay networks).

Important: Riverbed recommends that you enable HS-TCP only after you have carefully evaluated whether it will benefit your network environment. For detailed information about the trade-offs of enabling HS-TCP see, **tcp highspeed enable** in the *Riverbed Command-Line Interface Reference Manual*.

To display HS-TCP settings, use the CLI command **show tcp highspeed**. To configure HS-TCP, use the CLI command **tcp highspeed enable**. Alternatively, you can configure HS-TCP in the Management Console.

For details, see the *Riverbed Command-Line Interface Reference Manual* or the *Steelhead Appliance Management Console User's Guide*.

MTU Sizing

This section describes how Steelhead appliances work with PMTU Discovery and references RFC 1191 to negotiate Maximum Transmission Unit (MTU) sizing. This section includes the following topics:

- [“MTU Issues” on page 446](#)
- [“Determining MTU Size in Deployments” on page 446](#)
- [“Connection-Forwarding MTU Considerations” on page 447](#)

The MTU specifies the largest datagram packet (Layer-3 packet) that a device supports. In Steelhead appliance optimized environments, MTU sizing is typically automatic and not a concern. The default MTU size for a Steelhead appliance is 1500 bytes. This is the standard for many client and networking devices, especially across WAN circuits.

For pass-through traffic for an in-path Steelhead appliance without RSP, the Steelhead appliance passes packets up to the supported packet size of the configured in-path MTU. The in-path MTU supports jumbo frame configuration. For 1-Gbps interface cards, the supported MTU is 9216 or 16110, and all 10-Gbps cards support 16110. For a full list of interface cards and their MTU support, go to <https://supportkb.riverbed.com/support/index?page=content&id=s14344>.

For optimized traffic, the Steelhead appliances act as a proxy. A separate inner TCP connection is established between Steelhead appliances, with a potentially different MTU size from the original client-to-server connection.

When a Steelhead appliance detects that a session can be optimized, it initiates a TCP session to the remote Steelhead appliance using the IP flag *don't fragment* with packet size up to the value configured in the interface MTU (default 1500 bytes). In line with RFC 1191, if a router or device along the TCP path of the session (possibly originating a GRE tunnel) does not support the packet size, and because it is not allowed to fragment the packet, it can request the originator (the Steelhead appliance) to reduce the packet size. It does this with an ICMP type 3, code 4 (34) packet that carries the desired maximum size and the sequence number of the packet exceeding the router's interface MTU.

A common reason devices support less than 1500 bytes is the presence GRE tunnels used to establish VPNs. The 24-byte overhead GRE incurs effectively gives the tunnel interface a MTU of 1476 bytes.

Similar to the Path MTU Discover (PMTUD) behavior for most clients and servers, the Steelhead appliance reduces the packet size for a given session after it receives the ICMP message from the device with the lower MTU. According to RFC 1191, Section 6.3, the Steelhead appliance tries to send larger packets every ten minutes. For details, see <http://www.faqs.org/rfcs/rfc1191.html>.

In environments that support PMTUD, Riverbed recommends that you leave the Steelhead appliance MTU configuration to its default setting of 1500 bytes.

Note: In environments that support PMTUD, use the command `ip rt-cache rebuilt-count 0` on communicating Steelhead appliances (RiOS v8.0 and later).

For information about MTU and path selection, see [“MTU and MSS Adjustment When Using Firewall Path Traversal” on page 165](#).

MTU Issues

In most cases two hosts dynamically negotiate path MTU. Networks that contain firewalls or tunnels (VPN, GRE, IPsec transport mode) between Steelhead appliances sometimes require manual tuning of the MTU values. Firewalls and tunnels interfere in the following ways:

- Firewalls can contain rules that explicitly prevent path MTU by blocking or not sending the ICMP type 3 packets, causing all attempts at dynamically negotiating MTU to fail.
- Tunnels require additional per-packet overhead to encapsulate data, reducing possible MTU size for connections being carried.

Steelhead appliances set the DF bit for inner channel communication to peer Steelhead appliances. If the device in the network path does not support the Steelhead appliance packet size and also does not send an ICMP type 3 to notify the Steelhead appliance to reduce packet size, the packet is dropped without the Steelhead appliance knowing to reduce future packet sizes. This can result in poor optimization performance.

Determining MTU Size in Deployments

A simple method of determining MTU size across a path is to send *do not fragment* ping requests from the client PC, or client-side Steelhead appliance, with varying packet sizes to a remote server or Steelhead appliance. The following procedure shows the method from a windows client.

In the following example, a ping with the don't fragment (-f) and length (-l) 1500 bytes is sent to the remote server or Steelhead appliance. This results in 100% loss with *Packet needs to be fragmented but DF set* in the reply. Pinging 10.0.0.1 with 1500 bytes of data:

```
C:\>ping -f -l 1500 10.0.0.1
```


Packet needs to be fragmented but DF set.
 Packet needs to be fragmented but DF set.
 Packet needs to be fragmented but DF set.
 Packet needs to be fragmented but DF set.

Ping statistics for 10.0.0.1:

Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

Decrease the size of the packet to 1400 and repeat, pinging 10.0.0.1 with 1400 bytes of data:

```
C:\> ping -f -l 1400 10.0.0.1
```

```
Reply from 10.0.0.1: bytes=1400 time=222ms TTL=251
```

```
Reply from 10.0.0.1: bytes=1400 time=205ms TTL=251
```

```
Reply from 10.0.0.1: bytes=1400 time=204ms TTL=251
```

```
Reply from 10.0.0.1: bytes=1400 time=218ms TTL=251
```

Ping statistics for 10.0.0.1:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milliseconds:

Minimum = 204ms, Maximum = 222ms, Average = 212ms

This command gets the desired result. Repeat the ping test increasing or decreasing in increments of 10 or 20 until reaching the optimum value.

Note: Packet size of 1400 is shown only as an example, typical values can range from 1280 and higher.

When you specify the **-l <size>** command on a Windows machine, you are actually specifying the data payload, not the full IP datagram, including the IP and ICMP headers. To calculate the appropriate MTU, you must add the IP header (20 bytes) and ICMP header (8 bytes) to the Windows ping size. In the example of 1400 byte payload sent, the Steelhead appliance in-path MTU should be set on the Steelhead appliances to 1428 bytes. Note that although specifying ping sizes on Cisco routers, the specified size includes the IP and ICMP header (28 bytes). If using a Cisco device to test, set the MTU to the specified size, and adding the 28 bytes manually is not necessary.

How to configure the in-path MTU value

1. Go to Configure > Networking > In-path x_x
2. Change the MTU value and apply the setting.

In RiOS v8.0 and later, the Steelhead appliance does not pass through packets larger than the MTU value of its interfaces, nor does it send ICMP notifications to the sending host of the dropped packets. In environments in which the in-path MTU is lowered to account for a smaller MTU in the WAN network, Riverbed recommends that you use the command **interface mtu-override enable**.

Connection-Forwarding MTU Considerations

In Networks in which Steelhead appliances are connection-forwarding neighbors, it is critical that you configure the LAN or WAN links that are expected to carry forwarded traffic so they can support the configured in-path MTU. Connection-forwarded traffic does not support PMTUD.

When forwarded packets are too large, ICMP Type 3, Code 4 messages are generated on intermediate routers and sent back to the sending client or server. The ICMP header does not match a TCP connection in the client or server, which causes poor optimization or failed connections. To prevent poor optimization or failed connections, make sure that you configure the interface MTUs on links carrying forwarded traffic the same size as the Steelhead appliance in-path interface.

CHAPTER 22 Steelhead Mobile Deployments

The chapter describes Steelhead Mobile deployment. This chapter includes the following sections:

- “Overview of Steelhead Mobile Deployment” on page 449
- “Multiple Mobile Controller Deployments” on page 452
- “Ports Used with Mobile Controllers and Mobile Clients” on page 456
- “Location Awareness” on page 457
- “SSL with Steelhead Mobile” on page 459
- “Steelhead Mobile Best Practices and Other Considerations” on page 462

This chapter requires that you be familiar with the *Steelhead Mobile Controller User’s Guide*.

If you are using a release previous to Steelhead Mobile v4.0, see earlier versions of the *Steelhead Appliance Deployment Guide* and the *Steelhead Mobile Controller User’s Guide* on the Riverbed Support site at <https://support.riverbed.com>.

Overview of Steelhead Mobile Deployment

Before you begin the installation and configuration process for Steelhead Mobile, you must select a network deployment. This section describes the Steelhead Mobile deployment options. This section includes the following topics:

- “Basic Setup for Deploying Steelhead Mobile” on page 449
- “Steelhead Mobile with VPN Deployments” on page 450
- “Steelhead Mobile with Firewall Deployments” on page 451
- “Branch Office and Remote Access Deployments” on page 452
- “Multiple Mobile Controller Deployments” on page 452

Basic Setup for Deploying Steelhead Mobile

Steelhead Mobile ships with default policies. You can install and deploy Steelhead Mobile without modifying the default policies, or you can modify them to suite your environment.

If your network environment requires the deployment of multiple Microsoft Installer (MSI) packages, create the packages you need before you deploy the default package.

To install Steelhead Mobile using the default *Initial* policy provided, deploy the MSI package named *Default*. The default MSI package installs the default policies.

For the basic steps for how to install and configure the Mobile Controller and how to deploy the default MSI package to the Mobile Client in your network, see the *Steelhead Mobile Controller User's Guide*.

Steelhead Mobile with VPN Deployments

When you deploy Steelhead Mobile components in environments with VPNs, make sure that you do not optimize the VPN tunnel. If the VPN tunnel uses TCP for transport, add a pass-through rule to the policy for the VPN port number connected to by the client. Depending on your deployment scenario, this rule might be the first rule in the list.

VPNs that use IPsec as the transport protocol do not need a pass-through rule.

You can configure Mobile Controller with a VPN as follows:

- In-path
- Out-of-path

Figure 22-1 shows a deployment where both the mobile employee and the branch office use the same in-path Steelhead appliance.

Figure 22-1. In-Path Steelhead Mobile Deployment and VPN Tunnel

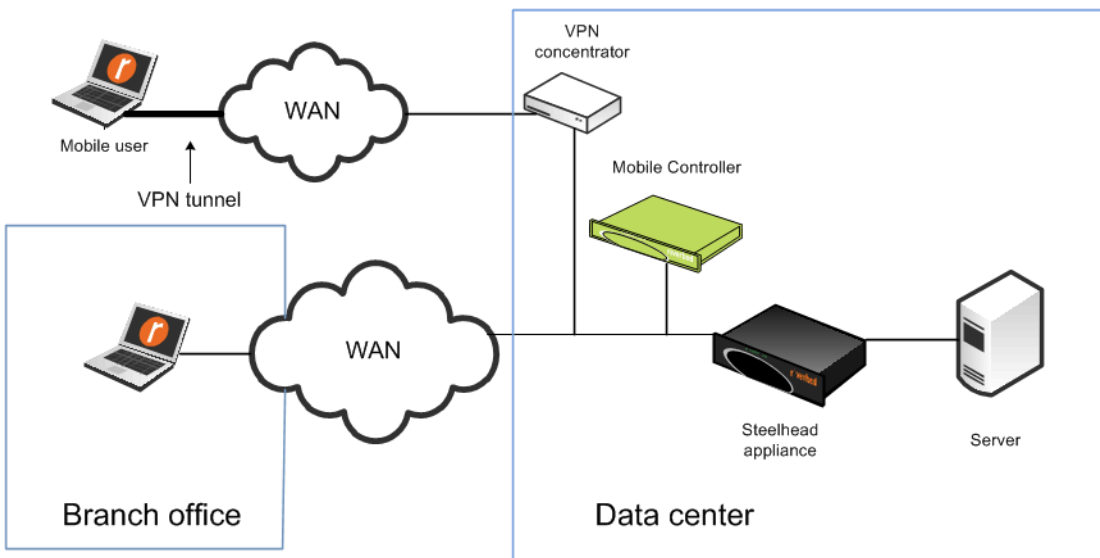
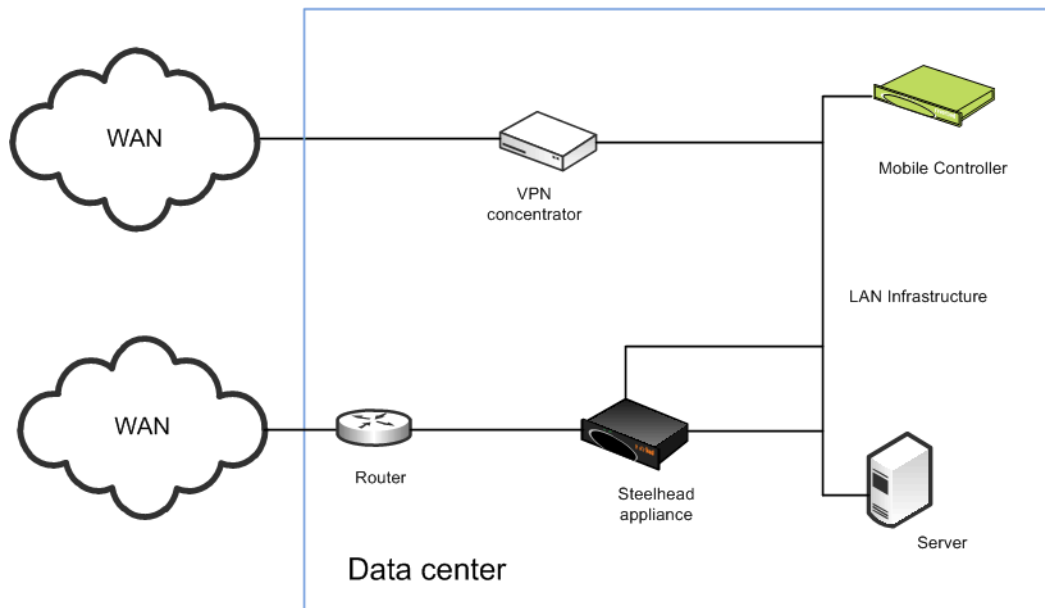


Figure 22-2 shows an in-path deployment where the mobile employee and the branch office use the same Steelhead appliance, but for the branch office Steelhead appliance is in-path; for the mobile employees, it is out-of-path.

Figure 22-2. Out-Of-Path Deployment Steelhead Mobile Deployment and VPN Tunnels



For more information about policies and pass-through rules, see the *Steelhead Mobile Controller User's Guide*.

Steelhead Mobile with Firewall Deployments

External firewalls, such as home firewall router appliances commonly found with broadband internet connections, do not require special settings for the Mobile Client when operating with VPN software on the client computer. The VPN software can have special requirements for external firewalls.

If you are using a firewall that does not allow outgoing connections, you must allow `rbtdebug.exe`, `rbtmon.exe`, `rbtsport.exe`, `rbtlogger.exe`, and `shmobile.exe`.

If you must access the Mobile Controller without the use of a VPN, both the client-side and server-side network firewalls must have some or all of ports 22, 80, 443, 7800, 7810, and 7870 open, as follows:

- Port 22 allows SSH access to the Mobile Controller from a remote site.
- Ports 80 and 443 allow Web access (including HTTP and HTTPS).
- Port 7800 is the default port between the Mobile Client and the remote Steelhead appliance for all optimized TCP sessions.
- You need to open Port 7810 on the network firewalls if you configure the Mobile Client to optimized connections with server-side out-of-path Steelhead appliances.
- The Mobile Client uses Port 7870 to send statistics to the Mobile Controller.

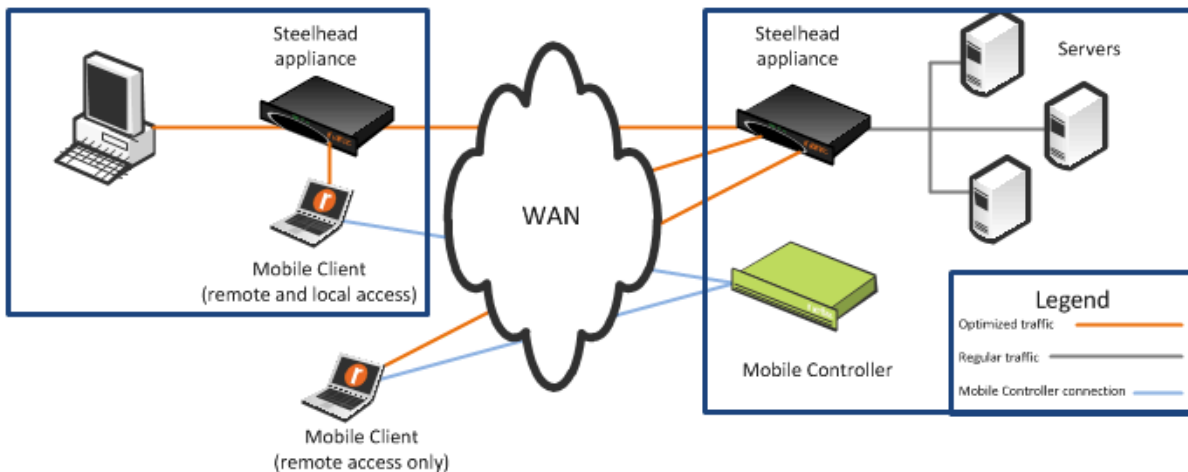
If you are using a VPN originating on the client machine, you do not need to open any of these ports mentioned previously.

Branch Office and Remote Access Deployments

In a branch office and remote access user deployment scenario, there are the following types of users:

- Local branch office users with systems that are already optimized by the local Steelhead appliance. These users do not need the Mobile Client software.
- Local branch office users who also remotely access the network. These users need the Mobile Client software, and their systems are optimized by the server-side Steelhead appliance.

Figure 22-3. Deploying Steelhead Mobile for Branch Office and Remote Users



If Mobile Clients are connecting to a branch office that already has a Steelhead appliance, you can enable enhanced auto-discovery on all Steelhead appliances. This allows the Mobile Client to bypass the local Steelhead appliance and optimize with the remote Steelhead appliance at the data center.

If you configure the branch warming feature in Steelhead Mobile v3.0 or later, the Mobile Client automatically detects the local Steelhead appliance when it is in the branch office (using location awareness). The Mobile Client does not consume a license when it is at the branch office. The Mobile Client continues to optimize with the remote Steelhead appliance, and also warms the Mobile Client RiOS data store, the local Steelhead appliance, and the remote Steelhead appliance.

For details on location awareness and branch warming, see [“Location Awareness” on page 457](#). For details on enhanced auto-discovery, see [“Peering Rules” on page 25](#) and the *Steelhead Appliance Management Console User’s Guide*.

Multiple Mobile Controller Deployments

This section describes the benefits of deploying more than one Mobile Controller, including deployment methods. This section contains the following topics:

- [“Overview of Multiple Mobile Controller Deployments” on page 453](#)
- [“Mobile Controller Concurrent User Limits” on page 454](#)
- [“Configuring Multiple Mobile Controllers for Redundancy” on page 454](#).
- [“Preparing to Join Mobile Controllers in a High-Availability Cluster” on page 456](#)

Overview of Multiple Mobile Controller Deployments

If you deploy multiple Mobile Controllers, you gain the following benefits:

- **Federation** - Different IT teams can manage designated areas.
- **Scale** - You can support a greater number of concurrently connected users.
There is a limit of concurrent user licenses per Mobile Controller. This limit is either 100 or 4000, depending on the Mobile Controller model. For details, see [“Mobile Controller Concurrent User Limits” on page 454](#).
- **Redundancy** - In case of a network outage or a failure of a Mobile Controller, users can continue to receive a license from another Mobile Controller and gain optimized access to network resources.
By default, when you deploy multiple Mobile Controllers, they operate as separate entities, with their own Mobile Client policies and concurrent user licenses. If you need identical policies across multiple Mobile Controllers, then you must individually update each instance. Concurrent user licenses on a failed Mobile Controller are not available for use by other Mobile Controllers.
- **High-availability cluster** - In case of a network outage or failure of a Mobile Controller, users can receive a license from another Mobile Controller and gain optimized access to network resources. If you configure a high-availability cluster, multiple Mobile Controllers automatically synchronize the Mobile Client policies among themselves. Concurrent user licenses on all Mobile Controllers in a cluster are pooled together as a single resource that is available to all Mobile Controllers in the cluster. To configure two or more Mobile Controllers as a high-availability cluster, the controllers must be running Steelhead Mobile v4.0 or later.

With more than one data center, you are not required to deploy multiple Mobile Controllers. If you have multiple data centers, but only one Mobile Controller, the Mobile Clients obtains a user license from the Mobile Controller no matter in which data center the Mobile Controller is located. The Mobile Controller is not directly involved with the optimized connections. Once the Mobile Client has a license, it optimizes connections with the Steelhead appliances in the data centers where the application servers are located.

It is important to distinguish between a deployment of multiple Mobile Controllers for redundancy, and multiple Mobile Controllers as a high-availability cluster. If you are using Steelhead Mobile v4.0 or later, Riverbed recommends that you use a high-availability cluster. The following table shows the differences.

Requirement	Multiple Mobile Controllers	Multiple Mobile Controllers in a High-Availability Cluster (Steelhead Mobile v4.0 or later)
Federated Steelhead Mobile	Yes	Yes
Autonomous Mobile Client policy management per Steelhead Mobile device	Yes	No
Global Mobile Client policy management	Partial - requires manual policy synchronization between Steelhead Mobile devices.	Yes
Ability to service license requests on failure of the Mobile Controller	Partial - requires sufficient additional endpoint licenses per Steelhead Mobile device.	Yes
Scalable beyond 4000 concurrent Mobile Client	Yes	Yes

Mobile Controller Concurrent User Limits

The following tables show the different Mobile Controller models and concurrent user limits.

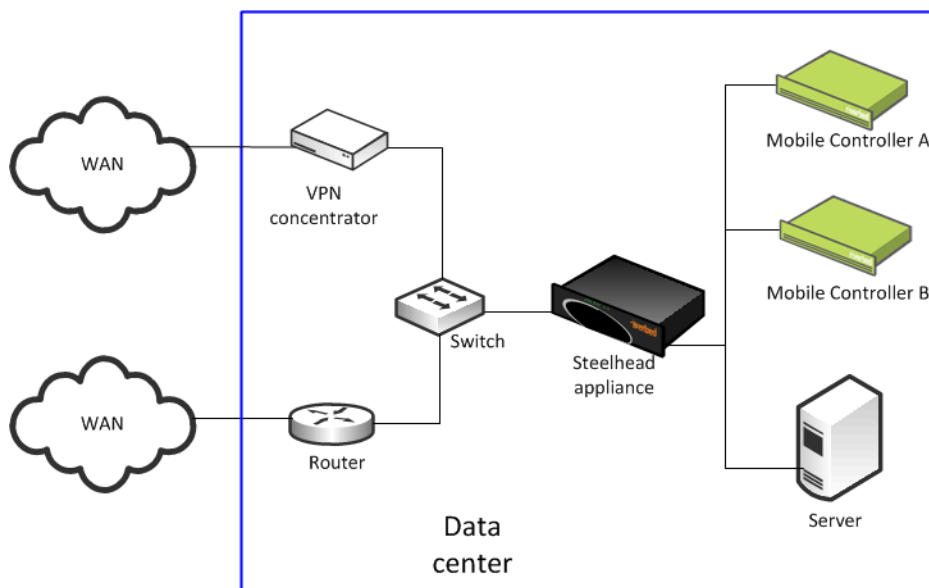
Mobile Controller Appliance	Default Number of Concurrent Users	Maximum Concurrent Users
SMC-8650-BASE	40	4000
Virtual SMC Licenses for VMware ESX Host and Steelhead EX v2.x	Default Number of Concurrent Users	Maximum Concurrent Users
SMC-VRT-V100	10	4000
SMC-VRT-V100-E	100	100
Virtual SMC Licenses for Steelhead EX v1.x	Default Number of Concurrent Users	Maximum Concurrent Users
VSMC-VSP	10	100
VSMC-VSP-E	100	100

Configuring Multiple Mobile Controllers for Redundancy

This section describes how to deploy two or more Mobile Controllers in a redundant configuration. This section requires that you be familiar with [“Multiple Mobile Controller Deployments” on page 452](#).

Figure 22-4 shows Mobile Controller A and Mobile Controller B installed in the data center and configured with a basic setup. You can install Mobile Controllers in different data centers and continue to provide redundancy.

Figure 22-4. Two Active Mobile Controllers



Complete the following steps on both of the Mobile Controllers.

To configure two Mobile Controllers for redundancy

1. From the Mobile Controller Console, choose Manage > Policies.
2. Select an existing policy to edit, or create a new policy.
3. Select Endpoint Settings.
4. Under Controller Options, add the hostname or IP address of each Mobile Controller.
To avoid any potential issues with invalid certificates when deploying in an SSL environment, make sure that the name and IP address in the Mobile Controller list is consistent with the hostname field in the Mobile Controller configuration.
5. Under Controller Options, select Use Random Ordering of Controllers when Connecting to ensure a random but even distribution of clients per Mobile Controller.

Figure 22-5. Endpoint Settings Page

The screenshot shows the 'Manage > Policies' page. At the top, there's a 'Default Policy Settings' section with a dropdown for 'Default Policy' set to 'Initial' and a 'Set Default' button. Below this are buttons for '+ Create New Policy' and '- Remove Selected Policies'. A table lists policies, with 'Initial' selected, showing its 'Last Modified Time' as '2012/06/21 15:32:21'. The 'Endpoint Settings' tab is active, displaying various configuration sections: 'General Settings' (with 'Show Client in the System Tray' checked), 'Data Store Settings' (with 'Data Store Size' at '10 GB'), 'Log Settings' (with 'Maximum Log Size' at '5000 kB' and 'Maximum Number of Log Files' at '2'), 'Controller Options' (with a table of controllers and 'Use Random Ordering of Controllers when Connecting' checked), and 'Windows-only Settings' (with two unchecked options).

Policy	Last Modified Time
Initial	2012/06/21 15:32:21

Controller Hostname	Port
192.168.0.102	7870
192.168.0.101	7870

You can repeat these steps to add additional Mobile Controllers to the redundant configuration later. In each case, add the hostname or IP address of the additional Mobile Controllers to the Endpoint Settings in the relevant policy on all the Mobile Controllers in the redundant configuration.

To avoid Mobile Clients receiving an untrusted certificate message, multiple Mobile Controllers in a redundant configuration must have the same Certificate Authority (CA) certificates. For more information, see [“Multiple Mobile Controllers and SSL” on page 461](#).

Preparing to Join Mobile Controllers in a High-Availability Cluster

This section describes how to prepare two or more Mobile Controllers to form a Mobile Controller high-availability cluster. Mobile Controller clusters work together as a single entity with respect to client policies, and some elements of the reporting. You can administer the Mobile Controller cluster from any member of the cluster.

Any client policy that you update on any cluster member automatically replicates and synchronizes with the other members of the cluster. Conflicts in which the same policy is inadvertently edited at the same time on different cluster members resolve automatically. Policies, packages, and some reporting statistics are automatically passed among cluster members. By default, this communication is through a TCP connection on port 7870.

Before a Mobile Controller can join a cluster, the Mobile Controller must:

- have a valid IP address.
- be able to ping other cluster members.
- be running Steelhead Mobile v4.0 or later.
- have its own set of licenses (CIFS, MAPI, concurrent user licenses, and so on).
- have the same signing CA certificate as the other Mobile Controllers already in the cluster.

There are not any geographic or latency restrictions between the nodes of a Mobile Controller cluster, because the Mobile Controllers in a cluster use the same protocol that a Mobile Controller uses with a Mobile Client. Typically, the current connection between the Mobile Controller and the Mobile Clients supports round-trip times in excess of three seconds.

After a Mobile Controller successfully joins the cluster, it is automatically populated with the packages, policies, and other client-related settings from the existing members of the cluster, and any existing settings are deleted. If you want to retain a copy of the original settings, make sure that you backup the Mobile Controller configuration before joining it to an existing cluster. You do not need to configure existing cluster members for a new Mobile Controller to join.

To set up a cluster, make sure that all Mobile Controllers meet the requirements listed above, choose one Mobile Controller to start with, and join the remaining Mobile Controllers to the cluster one-by-one.

For details on how to join the Mobile Controller to the cluster, see the *Steelhead Mobile Controller User's Guide*.

Ports Used with Mobile Controllers and Mobile Clients

Mobile Controllers and Mobile Clients use the following TCP ports to communicate with each other:

- **7870** - reporting, statistics, license requests, and policy pushes between Mobile Client and Mobile Controller

This is the default port for the communication between Mobile Controllers in a cluster.

- **80/443** - Mobile Client software upgrade
- **80/443** - Mobile Client upload of sysdump and tcpdump files using HTTP POST

Location Awareness

This section describes how to configure location awareness on the Steelhead appliance v6.0 or later, and the Steelhead Mobile v2.0 or later. This section includes the following topics:

- [“Overview of Location Awareness” on page 457](#)
- [“Branch Warming” on page 457](#)

Overview of Location Awareness

Location awareness enables Mobile Clients using Steelhead Mobile v2.0 or later to detect that they are in a branch office with a Steelhead appliance, and enables the branch office Steelhead appliance to optimize the Mobile Clients traffic.

When a Mobile Client is at a branch office that has a Steelhead appliance, location awareness enables you to choose which device performs optimization. If the Mobile Client performs optimization, the Mobile Client warms its local RiOS data store. It also consumes a Mobile Client license.

If a branch office and data center has a Steelhead appliance, you can use the Steelhead appliance for optimization so that the end user does not consume a Mobile Client license. But in that case, the Mobile Client does not warm its local RiOS data store. You can use branch warming to enhance location awareness to warm the Mobile Client RiOS data store, regardless of using a license. By default, branch warming is disabled.

For more information about branch warming, see [“Branch Warming” on page 457](#).

You can configure location awareness the following ways:

- **Adapter-based location awareness** - the Mobile Client optimizes traffic over only the adapters you specify. For example, most VPNs implement a virtual Ethernet adapter, but you can configure a rule to always optimize over VPN adapters and not over LAN adapters.

You cannot use adapter-based location awareness in:

- remote offices with a small number of users and no branch office Steelhead appliance.
 - hardware-based VPNs that do not terminate at the end-point or client.
- **Latency-detection location awareness** - enables the Mobile Clients to optimize data only if latency to the closest Steelhead appliance is greater than the threshold value. The default latency threshold value is 10 ms. To configure the value on the Mobile Controller, select Policies > Location Awareness.

Branch Warming

This section describes how branch warming on the Mobile Client interacts with the Steelhead appliances in the branch office and the data center. Branch warming requires the Mobile Controller and Mobile Client to run Steelhead Mobile v3.0 or later, and the Steelhead appliance to run RiOS v6.0 or later. Earlier versions of RiOS provide only location awareness, not branch warming.

This section includes the following topics:

- [“Branch Warming and Mobile Client Licenses” on page 459](#)
- [“Branch Warming and Enhanced Auto-Discovery” on page 459](#)

Branch warming works in conjunction with location awareness, enabling the Mobile Client user to experience warm acceleration regardless of the location. Branch warming tracks the data segments created while a Mobile Client is in a Steelhead appliance-enabled branch office. Location awareness enables the Mobile Client to detect that it is in a branch office with a Steelhead appliance, and enables the Steelhead appliance to optimize the Mobile Client traffic.

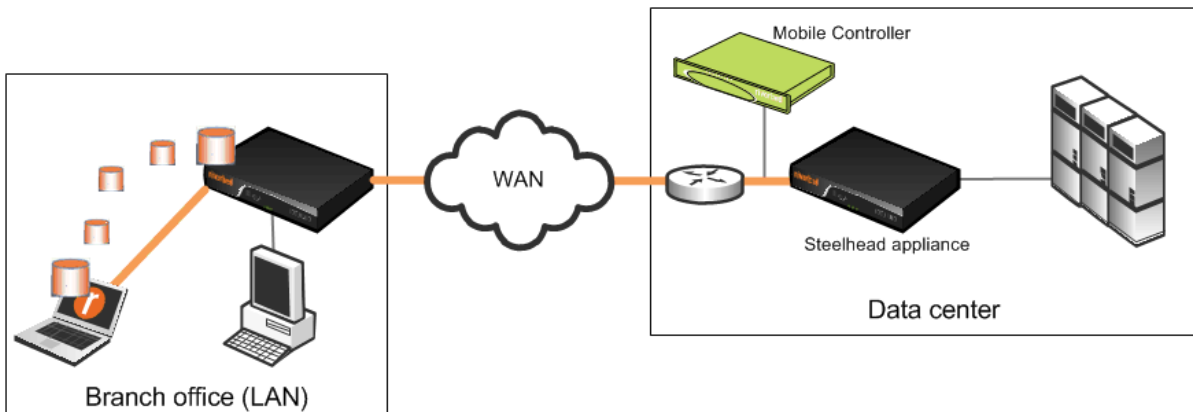
When you enable branch warming, the Mobile Client and the Steelhead appliance cooperate to provide warm data for out-of-branch use. The Mobile Client shares segments with the Steelhead appliance, thereby providing warm data wherever possible. Branch warming populates new data transfers that are occurring between the client and server, placing them into the RiOS data stores of the Mobile Client, the branch office Steelhead appliance, and the server-side Steelhead appliance.

When you download data from the server, the server-side Steelhead appliance checks to see if either the Mobile Client or the branch office Steelhead appliance has the data in its RiOS data store. If either device already has the data segments, the server-side Steelhead appliance sends only references to the data. The Mobile Client and the branch office Steelhead appliance communicate with each other to resolve the references.

Other clients at the branch office also benefit from branch warming, because data transferred by one client at a branch office populates the branch office Steelhead appliance RiOS data store. Performance improves for all clients at the branch office because they receive warm performance for that data.

Figure 22-6 shows how branch warming enables mobile users to optimize traffic with the server-side Steelhead appliance, while feeding segments that these users generate into the branch office Steelhead appliance RiOS data store.

Figure 22-6. Branch Warming Example



For each data request, the server-side Steelhead appliance checks whether the branch office Steelhead appliance or the Mobile Client RiOS data store making the request already has the data. If either one has the data, the Steelhead appliance sends a reference to the Mobile Client.

After the Mobile Client gets the reference, it checks if its RiOS data store already has the reference. If it does, the Mobile Client communicates with the server-side Steelhead appliance that it need not send the data again. Simultaneously, the Mobile Client checks whether the branch office Steelhead appliance has the same reference. If the branch office Steelhead appliance has the reference, the communication concludes; otherwise, the Mobile Client shares the reference and data with it.

If the Mobile Client does not have the reference or if its RiOS data store is deleted, it checks with the branch office Steelhead appliance to determine if it has the reference. If it does, then the Mobile Client takes the data segments from the branch-side Steelhead appliance and communicates with the server-side Steelhead appliance that it need not send the data again.

However, if the branch office Steelhead appliance does not have the reference, the Mobile Client requests the new data from the server-side Steelhead appliance and shares the new data and reference with the branch office Steelhead appliance so that at the end this communication, all three—the server-side Steelhead appliance, the branch office Steelhead appliance, and the Mobile Client—have the reference.

Branch Warming and Mobile Client Licenses

A Mobile Client with branch warming enabled, inside a branch office using the branch Steelhead appliance, uses one connection on the server-side Steelhead appliance and one connection on the branch office Steelhead appliance. It does not use a Steelhead Mobile license when in the branch mode. A single Steelhead Mobile license allows an unlimited number of connections.

The Mobile Client uses a license only when it detects that the Steelhead appliance with which it has optimized connections is not in branch mode.

Branch Warming and Enhanced Auto-Discovery

Enable enhanced auto-discovery on all Steelhead appliances to ensure that branch warming is successful. By default, enhanced auto-discovery is enabled.

With enhanced auto-discovery enabled, the last Steelhead appliance along the network path from the client to the server is automatically detected. Optimization then occurs between the Mobile Client and the last Steelhead appliance.

You can display, add, and modify enhanced auto-discovery settings on a Steelhead appliance in the Configure > Optimization > Peering Rules page.

SSL with Steelhead Mobile

SSL is a cryptographic protocol that provides secure communications between two parties over the Internet. This section includes the following topics:

- [“Traditional SSL Optimization” on page 460](#)
- [“Advanced High-Security SSL Optimization” on page 460](#)
- [“Configuring Steelhead Mobile and SSL” on page 461](#)
- [“Multiple Mobile Controller Deployments” on page 452](#)

Typically in a Web-based application, the client authenticates the server. You install an SSL certificate on a Web server for the client to check the credentials of the certificate to make sure it is valid and signed by a trusted third party. Trusted third parties that sign SSL certificates are called CA certificates.

Mobile Client v2.0 or later supports both traditional Steelhead appliance SSL optimization and advanced high-security SSL optimization. Riverbed recommends that you use advanced high-security SSL optimization to protect your system. You must have RiOS v5.5 or later installed on your Steelhead appliance to use advanced high-security SSL.

For more information about traditional Steelhead appliance SSL optimization, see the *Steelhead Appliance Deployment Guide - Protocols*.

Traditional SSL Optimization

In traditional Steelhead appliance SSL optimization with a Mobile Client, a client-side Steelhead appliance and the Mobile Client can optimize traffic from any client. The traditional SSL security mode enables clients to optimize SSL traffic to all Steelhead appliances before RiOS v5.5. RiOSs v5.5 or later can run in mixed deployments where one Steelhead appliance is running RiOS v5.5 and another Steelhead appliance in the network is running an earlier RiOS version.

In traditional SSL optimization, the client-side Steelhead appliance runs on the client machine. An attacker can redirect network traffic to their Mobile Client-enabled system and obtain the client session key sent from the server-side Steelhead appliance, thereby decrypting the client traffic.

To prevent such man-in-the-middle attacks, and to ensure that the Mobile Client can decrypt traffic originating on only one machine, Steelhead Mobile v2.0 or later provides advanced high-security SSL optimization.

Advanced High-Security SSL Optimization

This section provides an overview of advanced high-security SSL optimization. Riverbed recommends that you use advanced high-security SSL optimization to protect your system.

Advanced SSL optimization:

- enables the Mobile Client to optimize traffic from applications on the local system but not from any other system.
- requires RiOS v5.5 or later. If you are running a version earlier than RiOS v5.5, the Mobile Client supports the traditional Steelhead appliance SSL optimization.
- has specific browser requirements. For the most current requirements, see the release notes.

Figure 22-7. Advanced High-Security SSL Optimization Using a Mobile Client

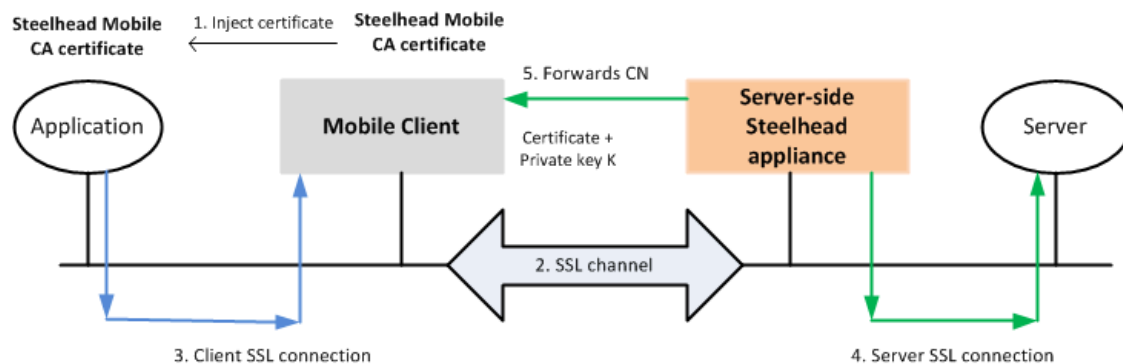


Figure 22-7 shows the steps in the advanced high-security SSL optimization using a Mobile Client. The steps are as follows:

1. The Mobile Client inserts a CA certificate into the trusted CA list using Internet Explorer or Firefox. The CA certificate is local to the Mobile Client.
2. When connections are initiated for SSL optimization, either on demand or proactively, the SSL inner channel is initiated.
3. The Mobile Client intercepts the client SSL connection from the client and terminates it.

4. The server-side Steelhead appliance connects to the server and extracts the common name (CN) from the server certificate. CN is the DNS name or IP address.
5. The server-side Steelhead appliance forwards the CN to the Mobile Client.
6. The Mobile Client takes the CN and uses the CA certificate it injected to generate a signed server certificate that it passes to the application. The application can trust this certificate because it is signed by the CA that exists in its trusted certificates list.

Configuring Steelhead Mobile and SSL

This section provides an overview of the basic steps required to configure SSL using Steelhead Mobile v4.0 or later and the Steelhead appliance v6.0 or later.

1. Obtain valid Enhanced Cryptography License Keys. This is required for every Steelhead appliance that peers with a Mobile Client and Mobile Controller.
2. Configure the server-side Steelhead appliance for SSL optimization with the SSL servers.
3. Configure a trust relationship between the Mobile Controller and the server-side Steelhead appliance.
 - Export the SSL certificate from the Mobile Controller.
 - Configure a proxy certificate and private key for the SSL back-end server on the server-side Steelhead appliance. This step enables the server-side Steelhead appliance to act as a proxy for the back-end server, which is necessary to intercept the SSL connection and to optimize it.
 - Import the Mobile Controller CA certificate into the server-side Steelhead appliance.
 - Export the server-side Steelhead appliance peering certificate.
 - Add the server-side Steelhead appliance peer to the Mobile Controller.

These mutual trust relationships establish secure inner channels between the Mobile Controller and the server-side Steelhead appliance.

For details on the secure inner channel, see the *Steelhead Appliance Deployment Guide - Protocols*.

4. Create or edit the policy on the Mobile Controller so that it allows the Mobile Client to intercept SSL connections.
5. Run a test to verify your configuration.

For details on configuring Steelhead Mobile and SSL, see the *Steelhead Mobile Controller User's Guide*.

Multiple Mobile Controllers and SSL

An SSL configuration requires there be a *trust relationship* between the Mobile Client and Mobile Controller. Without this trust relationship, the Mobile Controller cannot connect to and provide the Mobile Client with configuration details.

Mobile Controllers in a redundant deployment or a high-availability cluster must have identical signing CA certificates. Having identical signing CA certificates is a prerequisite for a Mobile Controller to join a cluster with other Mobile Controllers. Mobile Controllers generate their own certificates. If the Mobile Controllers do not have the identical signing CA certificates, the Mobile Clients receive an untrusted certificate message.

For more information about configuring trust relationships for Mobile Controllers and SSL, see the *Steelhead Mobile Controller User's Guide*.

Steelhead Mobile Best Practices and Other Considerations

This section lists best practices and includes other factors to consider when deploying Steelhead Mobile. This section includes the following topics:

- “Deployment Scenarios” on page 462
- “Management Best Practices” on page 463
- “Licensing Best Practices” on page 464
- “Antivirus Software” on page 464
- “Signed SMB Support” on page 465
- “Optimization Before User Log In” on page 465

Deployment Scenarios

Consider the following types of deployment scenarios. [Figure 22-8](#) shows the same Steelhead appliance for office and mobile employees—use fixed-target rules if the VPN ingress point is different.

Figure 22-8. Same Steelhead Appliance for Mobile Employees and Office Employees

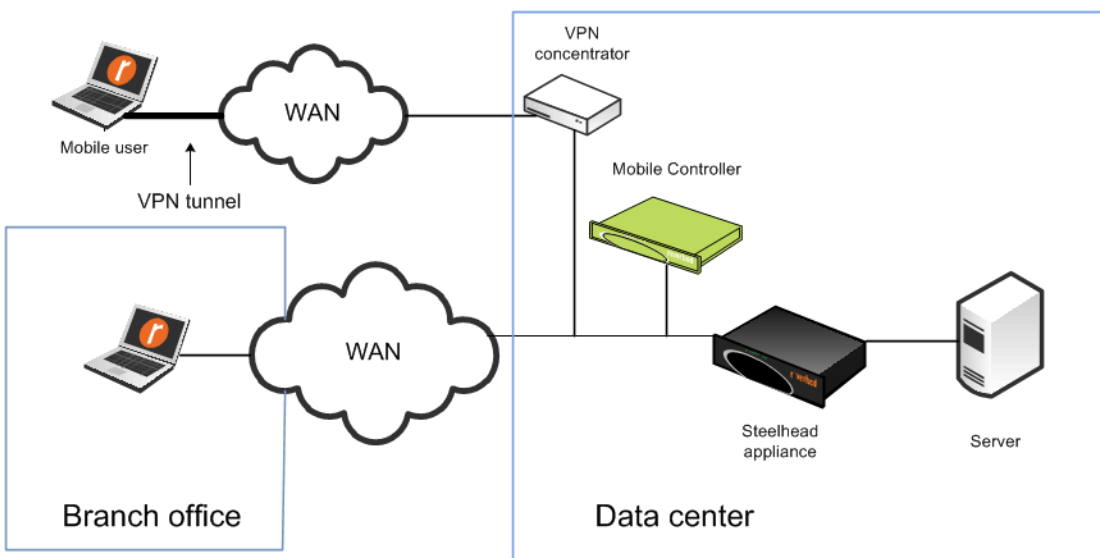
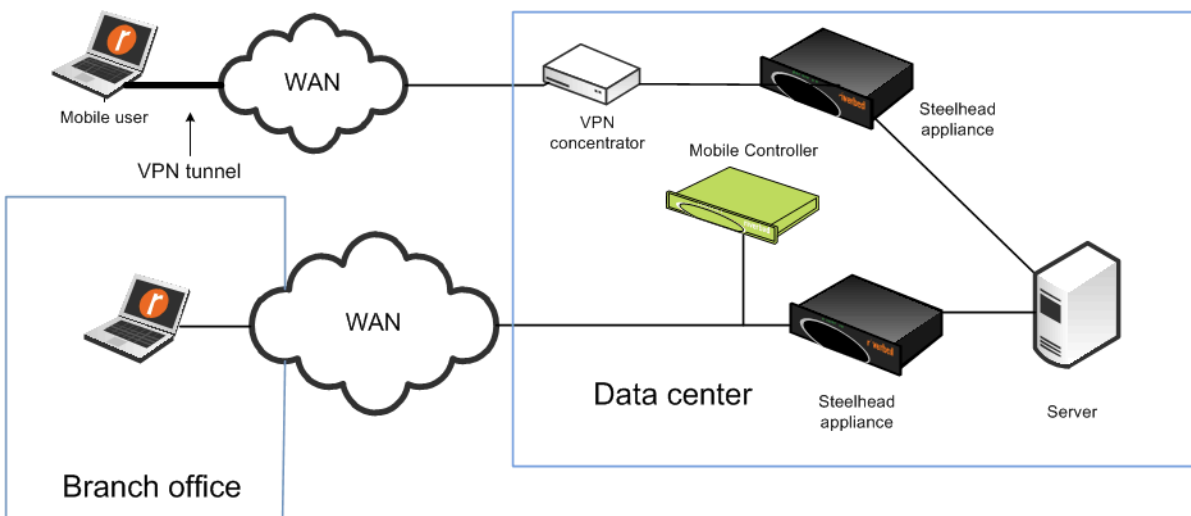


Figure 22-9 shows one Steelhead appliance for mobile employees and one Steelhead appliance for office employees.

Figure 22-9. Using One Steelhead Appliance for Mobile Employees and One Steelhead Appliance for Office Employees



Consider the following when deciding between using a Steelhead appliance or Steelhead Mobile for a branch office:

- Steelhead Mobile is best for small offices or offices with employees that spend the majority of their time out of the office.
- The Mobile Client has an individual RiOS data store, requiring software on each laptop. A Steelhead appliance has a shared RiOS data store.
- Steelhead appliances ensure that a resource is dedicated to acceleration.
- Steelhead appliances have features that are not available on Steelhead Mobile, such as RSP or VSP, prepopulation, and QoS.
- You can deploy Mobile Controller appliances in the data center for easier accessibility and connectivity.
- For installations where you expect to have fewer than 100 concurrent mobile users, consider installing Mobile Controller virtual edition using RSP or VSP on the Steelhead appliance in the data center.

Note: In RiOS v7.0.1 or later, RSP was replaced with VSP. VSP comes preinstalled in the Steelhead EX. For more information about VSP, see the *Steelhead EX Management Console User's Guide*. Your existing RSP packages work on VSP.

If you are deploying the Windows operating system and software by cloning, you might run into an issue in which you create duplicate SIDs (ghosting). For more information, see <http://supportkb.riverbed.com/support/index?page=content&id=S15558>.

Management Best Practices

The best practices to deploy and manage Steelhead Mobile are as follows:

- Understand your mobile user population by geography, client type, and division.

- Design systems that do not modify the default endpoint settings in the policy on the Mobile Controller.
If you decide to modify the default endpoint settings for groups of users, consider using fewer groups with more members, versus more groups with fewer members. For details on endpoint settings and policies, see the *Steelhead Mobile Controller User's Guide*.
- Use MSI packages to push Steelhead Mobile software to clients. MSI packages can also enable rollback or upgrade, ensuring easy maintenance.
- Riverbed recommends that you back up your Virtual SMC using CMC. For more information about backing up and restoring Virtual SMC, especially when running on ESXi in VSP on Steelhead EX see <https://supportkb.riverbed.com/support/index?page=content&id=S17487>
- Installation can be done in *visible* or *invisible* mode. In visible mode, end users have a Riverbed icon in the system tray that they can click for basic information and settings. In invisible mode, there are no visible icons to show that Steelhead Mobile is running on the end user machine.
- Do not use branch warming if you:
 - always want the client to optimize.
 - are using RiOS earlier than v6.0.

Licensing Best Practices

The best practices for end-user licensing are as follows:

- When considering licenses, do not count mobile employees who are behind the Steelhead appliance and are using branch warming.
- On the basis that not all mobile users are connected and active at the same time, estimate a 3:1 ratio of licensed versus connected users.
- A Mobile Client is issued a license by the Mobile Controller only after the Mobile Client initiates its first optimized connection request.
- For configurations in which you deploy multiple Mobile Controllers for high availability, use software v4.0 or later to allow pooling of end-user licenses from all the Mobile Controllers. Pooling of end-user licenses is an efficient use of the licenses if there is an unplanned Mobile Controller outage.

Antivirus Software

You can configure certain antivirus tools installed on a Windows or Mac platform to scan files that have recently changed. Configure the antivirus scanner to ignore the Mobile Client RiOS data store.

Because the Mobile Client is constantly updating its RiOS data store when the user is sending and receiving data with optimized connections, the Mobile Client RiOS data store is scanned frequently. This might lead to end-user performance problems. Because the Mobile Client RiOS data store does not contain files of any type (just unique data segments), there is no need to scan it for viruses.

Signed SMB Support

You do not have to configure the Mobile Client policy to support signed SMB connections. For traffic to optimize correctly:

- the Mobile Client must run v3.1 or later.
- you must configure the server-side Steelhead appliance to optimize signed SMB traffic (joined to a Windows domain, Transparent or Delegation mode enabled, and so on).

If configured correctly, the Current Connections report on the server-side Steelhead appliance shows CIFS-SIGNED.

For more details on signed SMB support, see *Steelhead Appliance Deployment Guide - Protocols*.

Optimization Before User Log In

As part of a Mobile Client installation, several processes run in the background, including *rbtsport*. Windows or MacOS starts *rbtsport* when the host operating system starts up. Therefore, even before a user has logged in, the optimization service is already running. This means that the Mobile Client can optimize a user login process. It can also provide optimization for other system processes that are occurring automatically in the background, such as backups.

Index

Numerics

802.1Q

PBR 257

trunk deployments 206

VLANs 206

A

ACLs, router command parameters for 247

Active and backup

Interceptor 352

N+M architecture 351

Adapter-based location awareness 457

Additional resources 3

Admission control 185, 417

Adobe, HTTP stream splitting 392

Advanced outbound QoS 88, 131

Alarm status

optimization service 424

Alarm status, optimization service 424

All active, N+M architecture 351

allow-failure command

connection forwarding and 44

fail-to-block and 200

Analyzer for NetFlow 420

Antivirus software

network slowdown and 437

old antivirus software 437

Steelhead Mobile 464

Apple HSL, HTTP stream splitting 392

Application Flow Engine 83

Lync 84

path selection 154

SSL common name matching 142

Application slowdown 432, 437

Application streamlining, overview 13

Asymmetric

network 434

routing, auto-detection of 58, 435

Authentication

features 398

using RADIUS 397

Auto in-path rule 25

Auto kickoff feature 26

Auto-detection of asymmetric routes 58, 435

Auto-discovery

enhanced auto-discovery 21, 33

firewall 21

IPv6 279

multiple in-path discovery 23

original auto-discovery process 19

overview 18

B

Bandwidth enforcement 100

Basic outbound QoS 85, 125

restrictions 86

WAN oversubscription 87

Basic steps to configure

NVS 329

path selection 157

physical in-path deployment 180

RADIUS 400

TACACS+ 403

virtual in-path, load balanced, L4

switch deployment 218

WCCP 229

Best practices

for Brocade 7500 377

for Brocade 7800 371

for Cisco MDS FCIP 367

for data streamlining and

compression 347

for deploying Steelhead appliances 36

for determining redirection and return

method 227

for enabling Steelhead appliance

security 410

for packet mode optimization 295

for policy control 413

for primary interface 259

for QoS 105

for SAN-to-Steelhead TCP/IP

connectivity 366

for securing access to Steelhead

appliances 404

for security monitoring 413

for Steelhead Mobile 462

for using ip wccp redirect exclude

command 231

Bi-directional synchronized RiOS data

store 9

Brocade 7500

best practices 377

Brocade 7800

best practices 371

Bypass interface card 173

C

- Cable and duplex
 - chain of in-path 178
 - overview 177
 - troubleshooting 179
- Cables, which type to use 177
- CDP
 - enabling on the Steelhead appliance 256
 - overview 256
- CIFS
 - default registry parameters 441
 - disabling SMB signing 441
- Cisco MDS FCIP
 - best practices 367
- Citrix
 - advanced outbound QoS 131
 - basic outbound QoS 125
 - inbound QoS 127
 - SSL 130
- Client request 434
- Clustering 39, 184
- Compression 347
- Configuring
 - a L4 switch 218
 - fake index feature 220
 - full address transparency for a VLAN 54
 - load balancing 218
 - NetFlow 220
 - PFS 387
 - RADIUS 399, 400
 - SSL 461
 - TACACS+ 402
 - transparent addressing 51
 - WAN visibility modes 72
 - WCCP 229
 - WCCP, specific redirection 246
- Connection forwarding
 - asymmetric networks and 435
 - failure handling with 44
 - IPv6 284
 - MTU 447
 - multi-interface support 44
 - neighborhood latency 45
 - sample configurations 198
- Connection limit and QoS 92
- Connection limits
 - OID and traps 417
- Connection pooling 12
- Controlling optimization 24
- Correct addressing
 - configuring 72
 - overview 50
- Crossover cables, overview 177

D

- Data protection
 - common deployments 349
 - overview 339
 - predeployment questionnaire 342
 - SnapMirror 353
 - third-party interoperability 355
- Data store, RiOS 8
 - alarms 15
- Data streamlining
 - best practices 347

- overview 8

- Data, synchronization of 434
- Default in-path rule 25
- Deny in-path rule 25
- Deployment example
 - in-path redundancy and clustering 184
 - IPv6 280
 - master and backup 184
 - multiple WAN router 192
 - multiple WAN router without connection forwarding 194
 - multiple WAN routers with connection forwarding 198
 - out-of-path 337
 - passing through transit traffic 30
 - path selection 159
 - PBR, with L2 switch and VLAN 260
 - PBR, with L3 switch 262
 - PBR, with Steelhead appliance connected to router 259
 - physical in-path with dual links 182
 - physical in-path, simple 180
 - serial cluster 187
 - serial cluster with multiple links 183
 - using in-path and peering rules 28
- Deployment modes for the Steelhead appliance 17
- Deployment options
 - for local and remote access users 452
- Design validation, path selection 167
- Direct SAN-to-Steelhead TCP/IP connectivity
 - best practices 366
- Discard in-path rule 25
- Disk performance 417
- Document conventions 3
- Documentation, contacting 5
- Dropped packets 437
- DSCP
 - QoS 78
 - QoS marking 121

E

- Enhanced auto-discovery 21, 33
 - Steelhead Mobile, branch warming 459
- Exclusive access 444

F

- Fail-to-block mode
 - allow-failure and 45
 - bypass cards for 175
 - connection forwarding and 200
 - detailed description of 175
 - enabling 176
 - in-path network interface cards and 41
 - introduction to 174
 - overview 41
- Fail-to-wire mode
 - cables for 36
 - effect on connected devices 175
 - introduction to 174
 - overview 41
- Failure detection in peers 190
- Failure modes
 - checking mode status of 176

- default setting for 41
 - fail-to-block 175
 - fail-to-wire 174
 - overview 174
 - transition notification of 174
- Fat pipe 445
- FCIP
 - profiles 367
 - storage optimization 358
 - tunnels on brocade 7500 377
 - tunnels on Cisco MDS 368
- FCIP Tunnels on Brocade 7500 377
- File shares, synchronization of 434
- Files, ensuring continuous access to 434
- Firewall
 - auto-discovery 21
 - path selection 164
 - stateful firewall 22
 - TCP probe 21
- Fixed-target in-path rules
 - IPv6 280, 286
 - overview 25, 33
 - to a primary address 35
 - to an in-path IP address 34
 - to primary IP address 337
 - WCCP router CPU spike and 438
- Flat mode 91
- Four-port appliance, as solution to network
 - asymmetry 436
- Full address transparency
 - configuring 72
 - disabling 73
 - overview 53
 - VLANs and 54
- Full address transparency with forward
 - reset 55
- G**
- Global DSCP, QoS 81
- GRE encapsulation method
 - avoiding use of 226
 - overview 226
- GRE encapsulation, path selection 153
- Guaranteed bandwidth and QoS 92
- H**
- Hierarchical Fair Service Curve 76
- Hierarchical mode
 - overview 88
 - traffic control process for 89
- High availability deployments
 - master and backup deployments 184
 - Mobile Controller 453, 456
 - multiple WAN router 192
 - parallel deployments 192
 - PBR 265
 - RiOS data store synchronization and 234
 - serial cluster 187, 192
 - serial cluster deployments 187
 - serial deployment 194
 - Steelhead Mobile 453, 456
 - WCCP and 234
- High-Speed TCP 348
 - as solution to underutilized fat pipe 445
- Host labels, QoS 141
- HSRP, multiple WAN router 192
- HTTP prepopulation 395
- HTTP stream splitting 392
 - Adobe 392
 - Apple HLS 392
 - Microsoft Silverlight 392
 - video 392
- Hybrid deployment 218
- I**
- Inbound QoS 98, 127
- Inbound redirection, as solution to router
 - CPU spike after WCCP configuration 438
- In-path default gateway and routing 173
- In-path deployments, overview 171
- In-path interface, IP address for 173
- In-path routes, as solution to packet
 - ricochet 437
- In-path rules
 - IPv6 280
 - types of 24
- In-path0_0 interface, overview 173
- Interceptor appliance 352
 - virtual in-path 218
- Interfaces
 - in-path IP address for 173
 - in-path0_0 173
 - primary 173
 - QoS and multiple WAN interfaces 105
- Intermediary
 - hub 433
 - switch 433
- IOS compatibility, and router CPU spike
 - after WCCP configuration 438
- IP address, for in-path interface 173
- ip wccp router command
 - how to use 244
 - overview 243
- IPv4, packet mode optimization 289
- IPv6
 - addressing 278
 - auto-discovery 279
 - connection forwarding 284
 - deployment scenarios 280
 - feature compatibility 276
 - fixed-target rules 280, 286
 - in-path rules 280
 - overview 275
 - packet mode optimization 289
 - protocol support 287
 - QoS 103
 - troubleshooting 287
- K**
- Kickoff feature 26, 175
- Known issues 4
- L**
- L2 method 226
- L4 switch, configuring 218
- LAN bypass 103
- Latency detection location awareness 457
- Latency priority 92
- Licensing, Steelhead Mobile 14, 464

- Link share weight 92
- Link state propagation 41, 176
 - overview 41
 - physical in-path appliance 177
- Live video. See HTTP stream splitting
- Load balancing, configuring 218
- Location awareness
 - adapter-based 457
 - latency detection 457
- Logical in-path deployment
 - L4 switch deployment, configuring 218
 - PBR, overview 218
 - WCCP, overview 218
- Logical in-path interface, overview 172
- Long fat pipe 445
- Lync and Application Flow Engine 84

M

- Management streamlining, overview 14
- Master and backup deployment 39
 - overview 184
 - timers 186
- Memory limits
 - OID and traps 419
- Microsoft Silverlight, HTTP stream
 - splitting 392
- Migration, QoS 107
- Mobile Controller
 - concurrent user limits 454
 - deploying multiple 452
 - high availability and SSL 461
 - high availability deployments 453, 456
 - multiple Mobile Controllers 454
- MTU sizing 445
 - connection forwarding 447
 - determining size 446
- Multiple WAN router
 - HSRP 192
- MX-TCP 93, 348
 - configuring QoS 135
 - N+M architecture 351
 - overview 93
 - transport streamlining 11

N

- N+M architecture 351
 - active and backup 351
 - all active 351
 - Interceptor appliance 352
 - MX-TCP 351
- NetFlow
 - configuring 220
 - PBR and 273
- Network
 - asymmetry 434
 - connection failure 437
 - connections 434
 - path 434
- Network interface card 173
- Nomenclature, path selection 151
- NSV 323, 326
 - basic steps 329
 - with VRF 325
- NTP 388
 - settings 413

O

- OID 417
 - connection limits 417
 - descriptions 421
 - memory limits 419
 - SNMP 421
- Online documentation 4
- OOB connection
 - destination transparency 70
 - full transparency 71
 - overview 69
 - path selection 156
- Opportunistic lock, symptoms of
 - unavailable 444
- Optimization
 - controlling with in-path rules 24
 - controlling with peering rules 25
 - overview 7
 - service alarm status 424
 - tools for controlling 24
- Outbound QoS (advanced), see also
 - advanced outbound QoS 88
- Outbound QoS (basic), see also basic out-
 - bound QoS 85
- Out-of-path deployments
 - clusters and 40
 - example of 337
 - limitations of 336
 - overview 335
 - QoS 104
 - redundancy and 40

P

- Packet mode optimization 289
 - best practices 295
 - configuring 290
 - does not support 295
 - IPv4 289
 - IPv6 289
 - supports 294
- Packet ricochet 45, 437
- Parallel cluster deployments
 - overview 198
- Parallel deployments, overview 192
- Path selection
 - Application Flow Engine 154
 - basic steps 157
 - design considerations 169
 - design examples 159
 - design validation 167
 - firewall path traversal 164
 - GRE encapsulation 153
 - nomenclature 151
 - OOB 156
 - overview 151
 - properties 152
 - QoS 152, 155, 159
 - site default path 152, 156
 - stateful security appliance 157
 - traffic candidates 154
 - virtual in-path 166
- Path, network 434
- PBR 330
 - 802.1Q 257
 - avoiding infinite loop with 255

- configuring 258
 - connecting appliances for 258
 - enabling CDP for 256
 - failover for 256
 - failover process for 256
 - High availability deployments 265
 - overview 218, 255
 - router CPU spike after WCCP 439
 - with L2 switch and VLAN 260
 - with Steelhead appliance connected to L3 switch 262
 - with Steelhead appliance connected to router 259
 - PCoIP, QoS 95, 143
 - Peering
 - peer failure detection 190
 - rules 25
 - Per-command accounting 398
 - Per-command authorization 398
 - PFS
 - as solution to unavailable files 434
 - broadcast mode 386
 - configuring 387
 - domain and local workgroup settings for 384
 - domain mode 384
 - local mode 386
 - local workgroup mode 385
 - models supported for 381
 - overview 381
 - share operating modes 386
 - share settings 388
 - share synchronization 383
 - shares, overview 383
 - shares, upgrading from v2.x 384
 - stand-alone mode 386
 - terms for 383
 - when to use 382
 - Physical in-path deployment 17
 - basic steps to deploying 180
 - examples of 180
 - failure modes for 174
 - master and backup 39
 - overview 171
 - redundancy options for 39
 - serial clusters and 40
 - Port transparency
 - configuring 72
 - disabling 73
 - overview 52
 - Port-based redirection 439
 - Prepopulation
 - HTTP 395
 - Primary interface, overview 173
 - Professional services, contacting 5
 - Proxy
 - certificate for SSL back-end server 461
 - Proxy file services 381
- Q**
- QoS**
- advanced outbound 88
 - Application Flow Engine 83
 - application, creating 116
 - applications, adding 120
 - bandwidth enforcement 100
 - basic outbound 85
 - basic outbound mode, site restriction 86
 - best practices for 105
 - branch office, configuring 119
 - Citrix and advanced outbound QoS 131
 - Citrix and basic outbound QoS 125
 - Citrix and inbound QoS 127
 - Citrix and SSL 130
 - class parameters for 92
 - class priorities for 93
 - class, default 88
 - classes, overview 88
 - configuring 109
 - connection limit parameter 92
 - data center, configuring 115
 - default-site 86, 139
 - DSCP 78
 - enforcement system for 91
 - enforcing policies using Riverbed QoS 78
 - FIFO 93
 - flat mode 91
 - global application list 85
 - global DSCP 81
 - guaranteed minimum bandwidth 92, 115
 - guidelines for Steelhead appliance models 102
 - hardware guidelines 101
 - hierarchical mode 88
 - host labels 141
 - inbound 98
 - inbound QoS limits 100
 - integrating Steelhead appliances into existing 78
 - integration techniques 79
 - IPv6 103
 - LAN bypass 103
 - latency priorities 92
 - list of remote sites 118
 - marking 80
 - marking and optimized traffic 113
 - marking default setting 81
 - marking on Steelheads 121
 - marking with DSCP 121
 - marking with TOS 121
 - migrating 107
 - multiple WAN interfaces 105
 - MX-TCP 93, 135
 - out-of-path deployments and 104
 - overview 76
 - path selection 152, 155, 159
 - PCoIP 143
 - performance 101
 - planning 109
 - queue types 93
 - Riverbed QoS 76
 - root class 88
 - rule requirements 100
 - rules 94
 - rules, maximum number allowed 100

- scheduler 90, 92
 - service policies, configuring on branch office 119
 - service policies, configuring on data center 116
 - service policy 86
 - service policy, modifying 120
 - SFQ 93
 - sites, maximum number allowed 100
 - SnapMirror 145
 - SSL common name matching 142
 - subnets, maximum number allowed 100
 - upgrading 108
 - upgrading marking 82
 - upper bandwidth parameter 92
 - virtual in-path deployments and 104
 - WAN oversubscription 87
 - WAN-side traffic characteristics 79
- R**
- RADIUS**
- CHAP 401
 - configuring 399, 400
 - fallback option 398
 - overview 397
 - per-command accounting 398
 - per-command authorization 398
 - remote and console method lists 399
- Redirect list 439
- Redirection and return methods
- best practices for determining 227
 - Cisco hardware that supports 226
 - overview 225
- Redundancy 39
- Release notes 4
- REST API 415
- Reversed mask redirection 225
- RiOS data store 8
- adaptive compression 347
 - bi-directional synchronization for 9
 - compression level 347
 - CPU settings 346
 - planning size for 17
 - size 417
 - synchronization of 15
 - synchronization requirements for 15
 - synchronization with HA deployments 234
 - unified 9
- Riverbed, contacting 4
- Router
- CPU spike 438
 - multicast groups 244
- RSP, Steelhead Mobile 463
- S**
- Sample network setup 326
- Satellite optimization 297
- bandwidth estimation 303
 - bit error rate 298
 - forward error correction 298
 - latency 298
 - SCPS 299
 - See also, SCPS
 - TCP automatic detect 305
 - TCP optimization 301
 - transmission errors 303
 - transport options 299
- SCPS 299
- compression 305
 - discovery 302
 - error tolerance 304
 - license 306
 - per connection transport 304
 - rate pacing 304, 311
 - SCPS negotiation. See SCPS discovery
 - single-ended rules 304
 - Steelhead 300
- SDR, overview 8
- SDRF
- selective optimization 364
- Selective optimization 364
- Serial cluster deployment 188
- Serial cluster deployments
- overview 187
- Server response 434
- Service policy
- QoS 86
- Share
- overview 383
 - settings, PFS 388
 - synchronization 383
- Simplified routing 45, 191
- as solution to packet ricochet 438
 - constraints 192
 - overview 45
 - simplified routing table 191
- Site default path
- path selection 152, 156
- SMB signing
- default registry parameters 441
 - disabling 441
 - overview 439
 - problem symptoms of 439
 - Steelhead Mobile 465
- SnapMirror
- data protection 353
 - optimization 353
 - QoS 145
- SNMP
- OID descriptions 421
 - supported versions 420
 - traps 422
- SNMPv3, authentication and privacy 426
- Spanning-tree, and fail-to-wire mode 175
- SRDF
- storage optimization 361
- SSL
- Citrix and QoS 130
 - configuring 461
 - managing certificates for web ui 414
 - overview with Steelhead Mobile 459
 - Steelhead Mobile high availability 461
- SSL common name matching 142
- Steelhead appliances
- choosing the right model 16
 - concurrent TCP connections 16
 - data store sizes for 17
 - deployment modes 17

- failure modes 174
- optimization process of 7
- QoS max sites, rules, and classes 101
- WAN bandwidth rating 17
- weight 223
- Steelhead Mobile
 - antivirus software 464
 - basic set up 449
 - best practices 462
 - deploying for branch office and remote access users 452
 - deploying in VPN environments 450
 - enhanced auto-discovery, branch warming 459
 - high availability deployments 453, 456
 - licensing 14, 464
 - overview with ssl 459
 - signed SMB support 465
 - transport modes 12
 - warm performance for 458
- Steelhead Mobile, virtual 464
- Storage optimization modules 358
 - FCIP 358
 - SRDF 361
- Straight-through cables, overview 177
- Subinterface, definition 325
- Symmetrix VMAX 362
- Synchronization
 - data 434
 - file share 434
- T**
- TACACS+
 - configuring 402
 - fallback option 398
 - first hit rejection 398
 - overview 397
 - per-command accounting 398
 - per-command authorization 398
 - remote and console method lists 399
- TCP
 - optimization with satellites 301
- TCP connections, concurrent connections in 16
- TCP IPv6 289, 290
- TCP probe, firewall 21
- TCP proxy mode 290
- TCP-PEP 297, 300, 302, 304, 305
- Technical Publications, contacting 5
- Technical support, contacting 4
- Third-party interoperability, data protection 355
- Timers
 - master and backup deployments 186
- TOS, QoS marking 121
- Traffic candidates, path selection 154
- Transparent addressing
 - compatible configurations for 51
 - configuring 72
 - firewalls between appliances and 59
 - full address transparency 53
 - full address transparency with forward reset 55
 - implications of 56
 - mis-routing optimized traffic 58
 - network asymmetry and 57
 - overview 51
 - port transparency 52
 - stateful systems and 56
- Transport streamlining 9
 - connection pooling 12
 - high-latency links 12
 - MX-TCP 11
 - overview 9
- Steelhead mobile TCP transport modes 12
- TCP algorithm selection 13
- TCP automatic detection 12
- WAN buffers 13
- Troubleshooting deployments 431
 - cable issues 179
 - IPv6 287
 - network asymmetry 434
 - old antivirus software 437
 - packet ricochets 437
 - router CPU spikes after WCCP configuration 438
 - satellite 315
 - SMB signing 439
 - unavailable opportunistic locks 444
 - underutilized fat pipes 445
- U**
- UDP IPv4 289, 290
- Unified data store 9
- Unoptimized
 - connections 434
 - files 444
- Upper bandwidth and QoS 92
- V**
- Video
 - Adobe HTTP dynamic streaming 392
 - Apple HLS 392
 - HTTP prepopulation 395
 - On-demand video. See HTTP prepopulation
 - HTTP stream splitting 392
 - live video. See HTTP stream splitting
 - Microsoft Silverlight 392
- Video optimization 391
- Virtual in-path
 - path selection 166
- Virtual in-path deployment 17
 - as solution to network asymmetry 435
 - clusters and 40
 - hybrid 218
 - overview 217
 - QoS 104
 - redundancy and 40
- Virtual Steelhead Mobile Controller 464
- VLAN
 - 802.1Q 206
 - full address transparency and 54
 - mapping 214
 - normalization 214
 - PBR 257
 - translation 214
- VLAN bridging 211
 - considerations 212

- L2 213
- L3 214
- multiple with VLAN mapping 214
- VRF 330
 - NSV 325
 - PBR 330
- VRF, with NSV 325
- VSP
 - see RSP

W

- WAN bandwidth rating 17
- WAN buffer 348
- WAN buffers, transport streamlining 13
- WAN capacity limit 416
- WAN disruption, inability to access files
 - during 434
- WAN oversubscription 87
- WAN visibility modes
 - configuring 72
 - correct addressing 50
 - full address transparency 53
 - overview 49
 - port transparency 52

WCCP

- access lists 246, 247
- ACLs for Steelhead appliances 248
- adding a Steelhead to an existing wccp
 - deployment 233
- additional features of 243
- assignment methods for 223
- basic configuration 229
- basic steps 229
- cluster 221
- clustering and failover 227
- clustering and failover for 227
- configuring 229
- enabling on the router 242
- fundamentals of 222
- GRE encapsulation method 226
- GRE, avoiding the use of 226
- group lists 245
- hash assignment 223
- high availability 227
- high availability deployment for 234
- ip wccp router command 243
- L2 method 226
- load-balancing and 249
- mask assignment 224
- multicast groups of 244
- NetFlow and 252
- overview 218, 221
- port-based redirection for 439
- pros and cons of 228
- redirection and return methods for 225
- reversed mask redirection for 225
- router configuration commands for 242
- router CPU spike after configuration
 - of 438
- service group password 243
- service group, overview 222
- simple deployment example of 230
- specific traffic redirection with 246
- traffic redirection and 242
- verifying configuration of 252

- WCCP cluster, definition 221