



## **HELIX PROXY ADMINISTRATION GUIDE**

Helix™ Proxy Version 13.0

Revision Date: 5 June 2009

RealNetworks, Inc.  
PO Box 91123  
Seattle, WA 98111-9223  
U.S.A.

<http://www.real.com>  
<http://www.realnexus.com>

©2003-2009 RealNetworks, Inc. All rights reserved.

Information in this document is subject to change without notice. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of RealNetworks, Inc.

Printed in the United States of America.

Helix, the Helix Logo, Real, the Real "bubble" (logo), RealJukebox, RealOne, Real-rTV, RealArcade, RealAudio, RealDownload, RealNetworks, RealPix, RealPlayer, RealPresenter, RealProducer, RealProducer Plus, RealPoducer Pro, RealProxy, RealPublisher, RealSites, RealSystem, RealText, RealVideo, Rhapsody, ra/ve, SureStream, The Future is Real, TurboPlay, and Xing are trademarks or registered trademarks of RealNetworks, Inc.

Other product and corporate names may be trademarks or registered trademarks of their respective companies.

# SUMMARY OF CONTENTS

INTRODUCTION.....	1
PART I: PROXY BASICS	
1 NEW FEATURES.....	9
2 OVERVIEW.....	13
PART II: CONFIGURATION	
3 INSTALLATION AND TESTING.....	29
4 PROXY SETUP.....	43
5 RATE CONTROL.....	61
6 PROXY ROUTING.....	97
7 MULTICASTS.....	103
8 MEDIA PLAYER CONFIGURATION.....	111
PART III: SECURITY	
9 FIREWALLS.....	119
10 ACCESS CONTROL.....	133
11 AUTHENTICATION.....	139
PART IV: MONITORING	
12 BASIC LOGGING.....	155
13 ADVANCED LOGGING.....	183
14 PROXY MONITOR.....	199
15 SNMP.....	203
PART V: APPENDIXES	
A CONFIGURATION FILE.....	219
B ADDRESS SPACE BIT MASKS.....	223
C AUTHENTICATION DATA STORAGE.....	229
GLOSSARY.....	235
INDEX.....	245



# CONTENTS

INTRODUCTION	1
What is Helix? .....	1
Audience for this Guide .....	1
How This Guide Is Organized .....	1
Conventions Used in This Manual .....	3
Terminology.....	4
Typographical Conventions.....	4
Sample Links.....	4
Default Locations and Values.....	5
Additional RealNetworks Resources .....	5
PART I: PROXY BASICS	
1    NEW FEATURES	9
New Features in Helix Proxy.....	9
Enhanced Rate Control.....	9
Server-Side Playlist Management .....	9
Removed Features in Helix Proxy Version 13.0 .....	10
Progressive Networks Audio (PNA).....	10
Support for MPEG-1 and Vivo Video Formats .....	10
Upgrade Issues .....	10
Compatibility with Earlier Versions .....	10
Rate Control Configuration Changes.....	10
ProcessorCount Variable Obsolete .....	10
Default Logging of Statistics Type 4.....	11
Binding Syntax for IPv6 Machines.....	11
Compatibility with Server Versions.....	11
Default Installation Directory.....	12
2    OVERVIEW	13
Introduction to Helix Proxy .....	13
Helix Gateway .....	13
Benefits of Using Helix Proxy .....	13
Media Formats.....	14
Protocols, Transports, and Packet Formats.....	15

- Media Player Configuration ..... 17
- Media Delivery Methods ..... 18
  - Accounting Connections ..... 18
  - Content Caching ..... 19
  - Pull-Splitting ..... 21
  - Pass-Through Delivery ..... 22
- Additional Features ..... 22
  - Helix Administrator ..... 22
  - XML-Based Configuration File..... 23
  - Bandwidth Restriction ..... 23
  - Proxy Routing ..... 23
  - Proxy Redundancy..... 23
  - Automatic Bandwidth Detection for Desktop RealPlayer ..... 23
  - Access Control ..... 24
  - Authentication ..... 24
  - Real-Time Monitoring ..... 25
  - Activity and Error Logs..... 25
  - SNMP..... 25
- Administering a Server and Proxy Together..... 25

PART II: CONFIGURATION

- 3 INSTALLATION AND TESTING ..... 29
  - Understanding Installation Issues ..... 29
    - Firewalls and Helix Proxy..... 29
    - Web Servers and Helix Proxy ..... 29
    - Installing Helix Gateway on a Single Machine..... 30
  - Installing Helix Proxy..... 30
    - Upgrading in a Different Directory..... 32
    - Reinstalling Helix Proxy in the Same Directory ..... 32
  - Running Helix Proxy..... 33
    - Starting Helix Proxy ..... 33
    - Stopping Helix Proxy ..... 36
  - Using Helix Administrator..... 37
    - Starting Helix Administrator ..... 38
    - Navigating the Interface..... 38
    - Helix Administrator Sections..... 39
    - Restarting Helix Proxy..... 41
    - License File Information..... 41
  - Testing Helix Proxy ..... 42
- 4 PROXY SETUP ..... 43
  - Modifying the Media Cache ..... 43

Changing Pull-Splitting .....	44
Pull-Splitting with RealSystem Server Version 8 .....	45
Defining Communications Ports .....	45
Shared UDP Port Ranges .....	46
Changing Streaming Media Ports .....	46
Changing the HTTP Port .....	47
Binding To An IP Address .....	48
Using Localhost .....	48
Capturing All Addresses .....	48
Modifying IP Addresses .....	49
Implementing Delayed Shutdown .....	50
Defining a Delayed Shutdown .....	50
Notes on Delayed Shutdown .....	51
Controlling Bandwidth .....	52
Setting UNIX User and Group Names .....	53
Setting Up Proxy Redundancy .....	54
Requirements for Using Redundant Proxies .....	55
Setting Up Redundant Proxies .....	55
Configuring Differentiated Services .....	56
Network Requirements for Differentiated Services .....	57
IP Header Bit Values .....	57
Configuring Differentiated Services .....	58
<b>5 RATE CONTROL .....</b>	<b>61</b>
Server-Side Rate Control Basics .....	61
Rate Control Basic Requirements .....	61
Types of Rate Control .....	62
Live Broadcasts and Rate Control .....	63
Media Formats Used with Server-Side Rate Control .....	63
Media Encoding Guidelines .....	64
User Agent Profiles .....	64
UAS Profiles .....	65
Configuration File Profiles .....	65
RDF Files .....	65
Profile Value Inheritance .....	66
User Agent Strings .....	67
Profiles Shipped with Helix Proxy .....	68
Streaming Rate Selection .....	70
Maximum Sending Rate .....	70
Initial Rate Selection .....	71
Initial Oversend Rate .....	72
Receiver and Sender Reports .....	72

Report Frequency .....	73
NADU Feedback .....	74
Receiver Report Timeout.....	74
Buffer Modeling .....	75
Data Size Versus Playback Time.....	75
Buffer Size.....	76
Preroll .....	76
Target Time .....	78
Downshifting .....	79
Downshift Depth Settings.....	80
Stepwise Downshifting .....	81
Client-Initiated Downshift .....	82
Streaming Below the Lowest Media Rate .....	82
Upshifting.....	82
Upshift Depth Settings .....	83
Stepwise Upshifting .....	84
Client-Initiated Upshift.....	85
Slowdown Mode .....	85
Slowdown Mode During Upshifting .....	85
Max Advance .....	86
Slowdown Mode Exit.....	87
Congestion Control.....	88
TFRC.....	88
BCC .....	89
TCP.....	89
Defining Rate Control Profiles.....	89
Using the Default Profile or Preset Values.....	94
Enabling BCC .....	96
<b>6 PROXY ROUTING .....</b>	<b>97</b>
Understanding Proxy Routing.....	97
Caching and Proxy Routing.....	98
Pull-Splitting with Proxy Routing .....	98
Pass-Through Delivery with Proxy Routing .....	99
Authentication with Proxy Routing.....	99
Setting Up Proxy Routing .....	100
Using Proxy Routing Rules .....	100
Defining Proxy Routing Rules .....	101
<b>7 MULTICASTS .....</b>	<b>103</b>
Understanding Multicasts.....	103
Media Players and Formats for Multicasting .....	104
Configuring a Network for Multicasts .....	105

	Multicast Addresses.....	106
	Packet Time to Live.....	106
	Multicasts with Multiple Network Interface Cards .....	107
	Address Requirements .....	107
	Configuring Back-Channel Multicasting.....	108
8	MEDIA PLAYER CONFIGURATION .....	111
	Manual and Automatic Configuration .....	111
	Configuring RealPlayers Manually.....	111
	Configuring Windows Media Players Manually.....	112
	Windows Media Player Versions 7 through 10.....	112
	Windows Media Player Version 11 and Later.....	113
	Configuring Automatic Proxy Redirection .....	113
	Media Player Requirements.....	114
	Network Switch Configuration .....	115
	Helix Proxy Configuration .....	115
PART III: SECURITY		
9	FIREWALLS .....	119
	How Firewalls Work.....	119
	Protocol Layers .....	120
	Transport-Layer Protocols.....	120
	Application-Layer Protocols .....	121
	Packet Formats .....	122
	Communicating with Media Players Behind Firewalls.....	123
	Control Channel.....	123
	Data Channel.....	123
	HTTP Cloaking.....	124
	Firewall Configurations.....	125
	Locating Helix Proxy Near the Firewall .....	125
	Working with Multiple IP Addresses .....	126
	Firewall Types.....	126
	Default Ports.....	129
	Media Players.....	129
	Origin Servers.....	131
	Helix Administrator .....	131
10	ACCESS CONTROL .....	133
	Understanding Access Control .....	133
	Rule Components.....	133
	Predefined Access Rules .....	134
	Access to Helix Administrator.....	134

	Access Rule Methods.....	134
	Rule Order.....	135
	IPv4 and IPv6 Access Rules .....	135
	Granting Access to Helix Administrator .....	136
	Creating General Access Rules.....	137
11	AUTHENTICATION .....	139
	Understanding Authentication .....	139
	Types of Authentication.....	139
	Media Requests Requiring Authentication .....	140
	Authentication Components.....	141
	Setting up Authentication .....	142
	Managing Users and Passwords .....	143
	Adding a User .....	143
	Deleting a User.....	144
	Browsing All User Names .....	145
	Changing a Password .....	145
	Using the Password Tool .....	146
	Using Databases .....	147
	Supported Database Types .....	147
	Adding a Database.....	148
	Setting Up Realms.....	149
	Authentication Protocols.....	149
	Creating or Modifying a Realm .....	151
PART IV: MONITORING		
12	BASIC LOGGING .....	155
	Understanding Basic Logging.....	155
	Basic Access Log .....	155
	Basic Error Log.....	156
	Log File Rolling .....	157
	Basic Access Log File Format.....	158
	Logging Style.....	158
	Access Log Fields.....	161
	Proxy Information .....	168
	GET Statements .....	169
	Client Statistics .....	171
	Statistics Type 1 .....	172
	Statistics Type 2 .....	173
	Statistics Type 3 .....	174
	Statistics Type 4 .....	175
	Customizing the Access and Error Logs.....	179

	Modifying the Basic Access Log .....	179
	Modifying the Basic Error Log .....	180
13	ADVANCED LOGGING .....	183
	Understanding Advanced Logging .....	183
	The Helix Proxy Registry .....	183
	Template Types .....	184
	Report Formats .....	185
	Using Session Templates .....	186
	Choosing a Watch Type.....	186
	Selecting the Output Format Type.....	187
	Defining Output Methods.....	187
	Console .....	187
	File .....	187
	HTTP Post .....	188
	TCP Broadcast .....	189
	UDP Broadcast .....	189
	Pipe and System Log on UNIX.....	189
	Windows NT Event Log.....	190
	Creating Logging Templates .....	190
	Sample Templates .....	193
	Using the Server Stats Templates .....	193
	Logging Proxy Configuration Changes .....	194
	Generating Client Statistics Reports.....	194
14	PROXY MONITOR .....	199
	Viewing Helix Proxy Activity .....	199
	Media Player Statistics .....	200
	Bandwidth Statistics .....	200
15	SNMP .....	203
	Understanding SNMP .....	203
	SNMP Plug-in.....	203
	Master Agent .....	203
	SNMP Protocol .....	204
	Management System and Management Information Base (MIB).....	205
	Configuring the SNMP Plug-In .....	205
	Configuring the Master Agent .....	207
	Modifying the Master Agent Configuration File .....	207
	Defining Master Agent Addresses and Ports .....	209
	Setting Up SNMP Security.....	209
	Defining a View Access Control Model.....	210
	Running the Master Agent on Windows.....	213

Starting the Master Agent on UNIX ..... 214  
Running a Management System ..... 214  
    Monitor Tree ..... 215  
    Configuration Tree ..... 215

PART V: APPENDICES

A CONFIGURATION FILE ..... 219  
    Understanding the Configuration File ..... 219  
    Editing the Configuration File ..... 220  
        XML Declaration Tag ..... 220  
        Comment Tags ..... 220  
        List Tags ..... 221  
        Variable Tags ..... 221  
        Helix Proxy Restart ..... 222  
B ADDRESS SPACE BIT MASKS ..... 223  
    Understanding Basic IP Address Construction ..... 223  
    Using a Bit Mask to Identify an Address Space ..... 223  
        Slash Notation ..... 224  
        Address Space Size ..... 224  
    Bit Boundaries ..... 225  
        Determining Bit Boundaries ..... 225  
    Working with 0-Bit and 32-Bit Masks ..... 227  
C AUTHENTICATION DATA STORAGE ..... 229  
    Understanding Authentication Data ..... 229  
    Using Text Files for Authentication Data ..... 229  
        Users Directory ..... 230  
        Logs Directory ..... 231  
    Using a Database for Authentication Data ..... 232  
        Users Table ..... 232  
        Access\_log Table ..... 232  
    Setting Up Other Types of Data Storage ..... 233  
        ODBC Databases ..... 233  
        mSQL ..... 234  
GLOSSARY ..... 235  
INDEX ..... 245

## INTRODUCTION

Welcome to Helix™ Proxy Version 13.0, the most powerful caching proxy server available for streaming media. Helix Proxy teams with media servers and players to optimize bandwidth and improve the playback experience. This manual will help you take full advantage of Helix Proxy for real-time delivery of media files.

### What is Helix?

Helix™ from RealNetworks is a universal digital media delivery platform. With industry-leading performance, integrated content distribution, advertising, user authentication, Web services support, and native delivery of RealMedia, Windows Media, QuickTime, and MPEG-4, Helix from RealNetworks is a robust digital media foundation that meets the needs of enterprises and networking service providers.

### Audience for this Guide

This guide is intended for system administrators who will set up and manage Helix Proxy. *Helix Proxy Administration Guide* is also available online at the following Web page:

<http://service.real.com/help/library/index.html>

### How This Guide Is Organized

This guide contains the following chapters.

#### Chapter 1: New Features

If you're familiar with previous versions of proxy servers from RealNetworks, this chapter will give you a quick update on the new features found in Helix Proxy.

**Chapter 2: Overview**

This chapter provides the “big picture” of how Helix Proxy works.

**Chapter 3: Installation and Testing**

Find out how to install and start Helix Proxy, and how to use the Web-based administration tool, Helix Administrator.

**Chapter 4: Proxy Setup**

This chapter discusses configuration options such as addresses, ports, and maximum bandwidth.

**Chapter 5: Rate Control**

The server-side rate control feature can vary the streaming rate of a clip that is encoded for multiple bandwidths. This chapter explains how to modify rate control for specific media players.

**Chapter 6: Proxy Routing**

By employing several Helix Proxies at once, you can funnel all streaming media Internet traffic through a single point.

**Chapter 7: Multicasts**

This chapter discusses multicasting, in which Helix Proxy relays a single, live stream to multiple media players, rather than a separate stream to each media player.

**Chapter 8: Media Player Configuration**

This chapter describes how to set up RealPlayer and Windows Media Player to contact Helix Proxy.

**Chapter 9: Firewalls**

If you are delivering content to users on the Internet, you’ll want to know how Helix Proxy and other RealNetworks products interact with firewalls.

**Chapter 10: Access Control**

Learn how to limit which media players use your Helix Proxy, based on their IP addresses.

**Chapter 11: Authentication**

Learn how to validate users attempting to access your Helix Proxy by requiring user names and passwords.

**Chapter 12: Basic Logging**

Helix Proxy can report media player behavior with a customizable degree of detail. Errors are reported in their own log, which can help you troubleshoot any problems that arise.

**Chapter 13: Advanced Logging**

Helix Proxy advanced logging provides a way for you to create a unique logging system using custom templates.

**Chapter 14: Proxy Monitor**

To provide highest quality service, you'll want to keep track of how many media players are accessing your Helix Proxy.

**Chapter 15: SNMP**

This chapter explains how to configure the Simple Network Monitoring Protocol (SNMP) plug-in and master agent to monitor Helix Proxy activity using third-party monitoring software.

**Appendix A: Configuration File**

This appendix discusses the XML-based configuration file used by Helix Proxy.

**Appendix B: Address Space Bit Masks**

This appendix explains how to identify a range of IP addresses by assigning a bit mask. This information is useful for the access control and multicasting features.

**Appendix C: Authentication Data Storage**

Helix Proxy comes with different methods for tracking authentication information, as described in this appendix. With this information you can set up your own authentication database.

## Conventions Used in This Manual

This section explains some conventional terms and formats used throughout the book.

## Terminology

- Because this guide is designed for the Helix Proxy administrators, the term *you* refers to the administrator. Those who play clips served by Helix Proxy are referred to as *visitors*, *viewers*, or *users*.
- Media players such as RealPlayer or Windows Media Player are referred to as *media players* or, more generically, as *clients*. Where information applies specifically to the RealNetworks® RealPlayer, this is clearly stated.
- The terms *clips*, *content*, *media clips*, and *media files* are used interchangeably to indicate the material that Helix Proxy streams.
- Production tools used to create the media clips that Helix Proxy streams are referred to collectively as *encoders*.

## Typographical Conventions

The following table explains the typographic conventions used in this guide.

### Notational Conventions

Convention	Meaning
<code>syntax</code>	This font is used for syntax of configuration files, URLs, or command-line instructions.
<i>variables</i>	Italic text represents variables. Substitute values appropriate for your system.
<b>emphasis</b>	Bold text is used for emphasis.
...	Ellipses indicate nonessential information omitted from examples.
[ ]	Square brackets indicate optional material. If you choose to use the material within the brackets, don't type the brackets themselves. An exception to this is in the access log, where statistics generated by the StatsMask variable are enclosed in regular brackets.

## Sample Links

Links that point to Helix Proxy take the following form:

helixproxy.example.com

where:

- `helixproxy` is a placeholder for the name of the computer on which you are running your Helix Proxy. Substitute the name of your organization's Helix Proxy computer wherever you see this syntax.
- `example.com` is a placeholder for a domain name. Substitute an actual domain name wherever you see this syntax.

## Default Locations and Values

In all of the examples given in this book, it's assumed that you've installed Helix Proxy in the default location for your operating system and that you're using default values for all settings. Of course, you can customize Helix Proxy however you want to meet your specific needs. Default values are used here for clarity of illustration. On Windows-based platforms, the default installation directory is:

`C:\Program Files\Real\Helix Proxy`

## Additional RealNetworks Resources

In addition to this guide, the following RealNetworks resources are available at: <http://service.real.com/help/library/index.html>

- *Helix Proxy Version 13.0 Release Notes*

The release notes contain the latest information about Helix Proxy. To view this information, click **Readme** in Helix Administrator.

- *Helix Proxy Configuration and Registry Reference*

This guide explains the configuration variables in the Helix Proxy configuration file. It also provides a reference for the registry variables that you can add to customized reports.

- *Helix Server Administration Guide*

This reference guide explains how to set up, configure, and run Helix Server to stream multimedia.

- *Helix Server Configuration and Registry Reference*

This guide explains the configuration variables in the Helix Server configuration file. It also provides a reference for the registry variables that you can add to customized reports.

- *Helix Server and Helix Proxy Troubleshooting Guide*

Refer to this document if you encounter problems while running Helix Proxy.

## PROXY BASICS

In this section, you learn about new features in Helix Proxy. If you're new to this technology, this section also gives you an overview of how Helix Proxy works.



## NEW FEATURES

This chapter discusses new features, removed features, and upgrade issues for Helix™ Proxy Version 13.0.

**Tip:** Before installing Helix Proxy Version 13.0, refer to the chapter on operating system modifications in *Helix Server and Helix Proxy Troubleshooting Guide*.

### New Features in Helix Proxy

Helix Proxy Version 13.0 introduces the following features.

#### Enhanced Rate Control

The Helix Proxy rate control mechanism now works with on-demand clips and live broadcasts encoded for multiple streaming rates. Rate control profiles used with Version 11 no longer function with the new rate control feature. However, Helix Administrator allows you to define rate control profiles for clients using a graphical interface.

**For More Information:** Refer to Chapter 5.

#### Server-Side Playlist Management

Helix Server Version 13.0 includes a server-side playlist management feature that allows it to stream a sequence of clips or broadcasts in a single RTSP session. Helix Proxy Version 13.0 supports playlist management in pass-through mode only.

**For More Information:** Refer to the chapter on playlists in *Helix Server Administration Guide*.

## Removed Features in Helix Proxy Version 13.0

Helix Proxy Version 13.0 does not include the following features, which were present in earlier versions of Helix Proxy.

### Progressive Networks Audio (PNA)

Helix Proxy has dropped support for the proprietary PNA protocol used in earlier RealNetworks client software versions. Previous versions of Helix Proxy supported PNA for compatibility with older RealNetworks clients (RealPlayer 5 and earlier).

### Support for MPEG-1 and Vivo Video Formats

Helix Proxy does not stream MPEG-1, MPEG-2, or the Vivo video formats. It continues support for MP3, as well as the MPEG-4 version of the MPEG standard.

## Upgrade Issues

This section explains issues about upgrading to Helix Proxy Version 13.0.

### Compatibility with Earlier Versions

Helix Proxy Version 13.0 is not compatible with configuration files used with Helix Proxy Version 11 and earlier. If you are upgrading to Helix Proxy Version 13.0 from any version other than Version 12, use the new configuration file created by the installation program.

### Rate Control Configuration Changes

The server-side rate control mechanism described in Chapter 5 was completely revised beginning with Helix Proxy Version 12. Rate control profiles for Helix Proxy Version 11 and earlier no longer function with Helix Proxy Version 13.0, and must be recreated.

### ProcessorCount Variable Obsolete

The new StreamerCount variable replaces ProcessorCount, which should be removed from the configuration file. Refer to the chapter on basic streaming features in *Helix Proxy Configuration and Registry Reference*.

## Default Logging of Statistics Type 4

By default, Helix Proxy records client statistics type 4 to its access log. The default in earlier versions was statistics types 1 and 2. If you wish to continue logging statistics other than type 4 statistics, you must modify the logging feature after you install Helix Proxy Version 13.0.

Note, too, that previous versions of Helix Proxy recorded only statistics type 4 if you selected statistics types 1, 2, and 4. This was because statistics type 4 included the same information found in statistics types 1 and 2. When you choose statistics types 1, 2, and 4 with Helix Proxy Version 13.0, however, you will log all three statistics fields.

**For More Information:** See “Customizing the Access and Error Logs” on page 179.

## Binding Syntax for IPv6 Machines

The older syntax to bind to all IP addresses on a machine, 0.0.0.0, binds only the IPv4 addresses on a dual-stack IPv4/IPv6 machine. To bind to all IPv4 and IPv6 addresses on a dual-stack machine, specify any as the binding option.

**For More Information:** See “Binding To An IP Address” on page 48 for an explanation of how to set bindings, along with a list of additional binding options.

## Compatibility with Server Versions

All media delivery features of Helix Proxy Version 13.0 are compatible with the latest version of Helix Server, as well as most earlier versions of RealSystem Server. The following table lists compatibility with server versions for the three methods that Helix Proxy uses to deliver content.

**Proxy Compatibility with Server Versions**

Helix Proxy Feature	Helix Server versions 9 and higher	RealSystem Server		
		versions 8 and 7	G2 (version 6)	version 5 and earlier
Pass-through	<b>yes</b>	<b>yes</b>	<b>yes</b>	<b>yes</b>
Pull-splitting	<b>yes</b> (RTSP only)	<b>yes</b> (RTSP only)	<b>yes</b> (RTSP only)	no
Caching	<b>yes</b>	<b>yes</b>	<b>yes</b>	no

**Note:** To use pull-splitting with RealSystem Server version 8, you need to modify the Helix Proxy configuration file. Refer to “Pull-Splitting with RealSystem Server Version 8” on page 45.

**For More Information:** For descriptions of pass-through, pull-splitting, and caching, refer to “Media Delivery Methods” on page 18.

## Default Installation Directory

On Windows, Helix Proxy installs into the following default location, which differs from the installation paths for earlier versions of RealSystem Proxy:

C:\Program Files\Real\Helix Proxy

If you choose the default location, you may need to move logs and other files to the new directory tree, as described in “Upgrading in a Different Directory” on page 32.

## OVERVIEW

Designed to re-serve streaming media securely, Helix Proxy delivers an impressive array of media to the broadest range of media players. This chapter introduces you to Helix Proxy concepts and features.

### Introduction to Helix Proxy

This section explains the benefits of using Helix Proxy, and introduces you to the media formats and protocols that Helix Proxy supports.

#### Helix Gateway

Helix Proxy is one component of Helix Gateway. The other component is Helix Server, from which content originates. Typically, you install Helix Proxy and Helix Server on separate machines. However, you can install them on the same machine, as described in “Installing Helix Gateway on a Single Machine” on page 30.

#### Benefits of Using Helix Proxy

Helix Proxy is software that you install on a network or ISP gateway to consolidate requests for media streamed from Helix Server. Typically residing behind a firewall, Helix Proxy provides several benefits:

- Helix Proxy reduces bandwidth consumption by eliminating redundant data transmissions. Instead of Internet-based media servers delivering all clips and live broadcasts, Helix Proxies can cache requested clips in its media cache, and split a single broadcast stream to multiple media players. By redistributing media data behind the firewall, you can greatly reduce the amount of bandwidth consumed by media streaming through the firewall.

**Note:** Additionally, Helix Proxy provides mechanisms for controlling inbound and outbound bandwidth parameters, thus securing bandwidth for other applications.

- Helix Proxy improves the quality of the user experience by distributing streaming media closer to the user. Because media streams travel a shorter distance from the cache to media players, the likelihood of network congestion, latency, and packet loss is reduced, improving quality of service.
- Helix Proxy allows you to enforce policies on media access through selective filtering and user authentication.
- Helix Proxy masks the IP addresses of the media players requesting media across the Internet.

## Media Formats

Helix Proxy can proxy every media type that Helix Server can serve. Helix Server streams on-demand clips and broadcasts live events in more media formats than any other media server. Depending on its license type, Helix Server can serve the following file formats:

RealNetworks:	RealAudio (.rm, .ra), RealVideo (.rm, .rmvb), RealPix (.rp), RealText (.rt)
Microsoft:	Windows Media (.asf, .wma, .wmv)
Apple:	QuickTime (.mov)
Standards-Based:	MPEG-4 (.mp4), MP3 (.mp3)

Working with Helix Server, Helix Proxy can deliver the same media formats on any platform, including Windows and many UNIX variants. This allows you to stream any supported format using the operating system of your choice. Helix Proxies running on different operating systems are completely interoperable, allowing you to place multiple proxies and servers in a heterogeneous network environment.

**For More Information:** For more on supported media formats, refer to the overview chapter of *Helix Server Administration Guide*. The specific versions of supported media formats are subject to change. For the latest information, check <http://www.realnetworks.com/resources>.

## Support for Non-Hinted MPEG-4 Tracks

Helix Proxy generally can stream from its cache MPEG-4 clips encoded with any codec, as long as the clips include a hint track. In addition, Helix Proxy can stream clips encoded with certain codecs whether or not the clips contain hint tracks. The following table lists the MPEG audio and video codecs that do not need to include hint tracks. For these codecs, Helix Proxy refers to the hint track if it is present, but still streams the media packets if the track is absent.

**MPEG-4 Codecs That Do Not Require Hint Tracks**

Codec	Type	MIME Type	Reference
H.263, PO, P3	Video	video/h263-2000	<a href="http://www.ietf.org/rfc/rfc2429.txt">http://www.ietf.org/rfc/rfc2429.txt</a>
MPEG-4 (SPL0)	Video	video/mp4v-es	<a href="http://www.ietf.org/rfc/rfc3016.txt">http://www.ietf.org/rfc/rfc3016.txt</a>
AAC	Audio	audio/mp4a-latm	<a href="http://www.ietf.org/rfc/rfc3016.txt">http://www.ietf.org/rfc/rfc3016.txt</a>
AMR-NB	Audio	audio/amr	<a href="http://www.ietf.org/rfc/rfc3267.txt">http://www.ietf.org/rfc/rfc3267.txt</a>
AMR-WB	Audio	audio/amr-wb	<a href="http://www.ietf.org/rfc/rfc3267.txt">http://www.ietf.org/rfc/rfc3267.txt</a>

**Note:** Through the configuration file, you can turn off hint track reading entirely for these codecs. For more information, refer to *Helix Proxy Configuration and Registry Reference*.

## Protocols, Transports, and Packet Formats

The following table summarizes the protocols, transports, and packet formats supported by Helix Proxy.

**Supported Protocols and Data Packet Formats**

Control Protocol	Data Packet Format	Data Packet Transport	Proxy Support?
RTSP	RDT (RealNetworks), RTP	IP multicast, UDP, TCP	<b>Yes</b>
MMS	Microsoft proprietary	UDP, TCP	<b>Yes</b>
HTTP	RDT, RTP	TCP	No

**For More Information:** For details about the control transports and data packet formats allowed on each port, see Chapter 9.

## IP Version 6

Helix Proxy supports both version 4 IP addresses (IPv4) and the newer version 6 IP addresses (IPv6). You can run Helix Proxy on a computer that uses one or more IPv4 addresses, as well as a machine that uses both IPv4 and IPv6 addressing. Note the following about using IPv6 addresses:

- Where possible, RealNetworks recommends that you specify DNS names instead of IP addresses. This allows Helix Proxy or another machine to resolve the domain name to an IPv4 or IPv6 address based on its network capabilities and the most efficient means of routing the request.
- Helix Proxy supports full IPv6 syntax, such as the following:

- 1080:0:0:0:8:800:200C:417A

It also supports the use of a double colon (“::”) to compress leading or trailing fields containing only zeroes, as in the following:

- 1080::8:800:200C:417A

- Helix Proxy supports only the Classless Inter-Domain Routing (CIDR) format for IPv6 bitmasks, as shown in the following example:

2001:638:a01:2::/64

**For More Information:** Many Web resources explain IPv6 CIDR notation. See, for example, [http://en.wikipedia.org/wiki/Classless\\_inter-domain\\_routing](http://en.wikipedia.org/wiki/Classless_inter-domain_routing).

- Helix Proxy can stream on-demand content or live broadcasts using any streaming protocol to any media player that supports IPv6. For media URLs, however, RealNetworks strongly recommends using domain names to allow streaming to clients that handle only IPv4.
- The section “Binding To An IP Address” on page 48 explains how to bind Helix Proxy to various IP addresses on a multi-stack machine.
- The access control feature allows you to define access rules for both IPv4- and IPv6-based client connections. The section “IPv4 and IPv6 Access Rules” on page 135 explains access rule checking for the two protocols.
- Helix Proxy does not support IPv6 addresses for the following features:
  - Multicasts, which Chapter 7 describes.
  - SNMP, which is covered in Chapter 15.

- Differentiated Services, described in the section “Configuring Differentiated Services” on page 56.

### Control Protocols

Helix Proxy primarily handles media player requests and streams content using Real-Time Streaming Protocol (RTSP), an Internet standard control protocol for streaming multimedia. Although Helix Server can stream through HTTP, Helix Proxy is not an HTTP proxy and does not handle any media requests made through HTTP between media players and an origin Helix Server.

**Note:** Helix Proxy can proxy MMS requests for older versions of Windows Media Player. However, Helix Proxy cannot proxy RTSP or HTTP requests for Windows Media content residing on Helix Server. For more information, refer to “Windows Media Player Version 11 and Later” on page 113.

### Transport Streams

Helix Proxy works with connecting media players such as RealPlayer and Windows Media Player to determine the best transport for a stream, whether IP multicast for live broadcasts, or UDP and TCP for broadcasts or on-demand content.

### Packet Formats

Media types streamed by Helix Server and Helix Proxy use two primary packet formats:

- RDT—a proprietary packet format native to RealNetworks
- RTP—an Internet standard data type packet format

## Media Player Configuration

For Helix Proxy to fulfill requests, media players such as RealPlayer and Windows Media Player must be configured to send their requests to Helix Proxy. You can configure media players to contact Helix Proxy directly, or you can configure your network to make Helix Proxy intercept media player requests. For more on these methods, refer to Chapter 8.

## Media Delivery Methods

Helix Proxy uses three different methods to stream clips to media players:

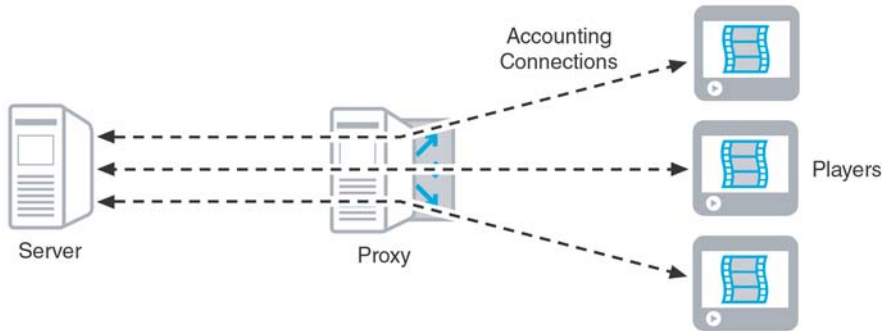
- **Content Caching**—For on-demand requests, Helix Proxy securely stores the streaming media data for later viewing by other media players.
- **Pull-Splitting**—For live requests, Helix Proxy “shares” the stream among the media players that request it. In addition, Helix Proxy can transmit the stream by using multicast.
- **Pass-Through Delivery**—Helix Proxy passes the stream from the originating server. No bandwidth conservation occurs.

Helix Proxy automatically selects the most efficient delivery method possible, based on the type of content requested and the network configuration.

## Accounting Connections

When a media player requests a clip or live broadcast, Helix Proxy forwards the request to the server on which the stream originates. Helix Proxy always opens this accounting connection between the media player and the origin server, regardless of the streaming method used.

### Accounting Connection Opened By Helix Proxy



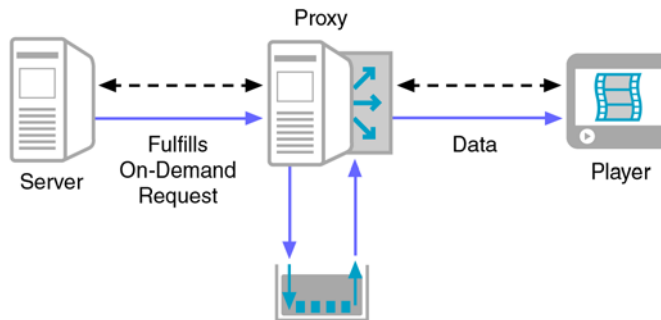
The origin Helix Server verifies the clip’s existence, and determines if the media player is authorized to receive the content. If the server denies the request, Helix Proxy complies with the denial, sending the media player an error message. Only after Helix Server has authorized the media player’s request will Helix Proxy begin streaming. An origin Helix Server may deny a request for the following reasons:

- The requested content may be secured. If the content requires user name and password validation, for instance, Helix Server does not allow streaming until it receives the correct user name and password combination.
- The IP address of the media player or Helix Proxy may be denied access based on the access control list maintained by Helix Server.
- Based on the licensed capacity or other bandwidth restrictions of Helix Server, no more connections may be available.

## Content Caching

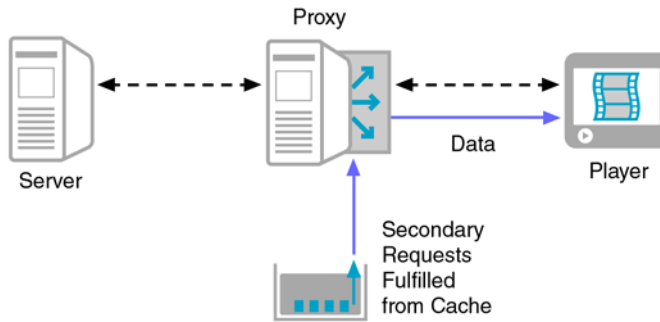
Caching of on-demand clips is the default behavior, and is turned on automatically in Helix Proxy. The first time a media player requests a media clip from Helix Server, Helix Proxy acquires and stores that clip, as illustrated in the following figure.

### Media Cache Filled with an On-Demand Clip



When a subsequent media player requests the same clip, Helix Proxy uses its accounting connection to check if a newer version of the clip exists on Helix Server. If it determines that the cached copy is up-to-date, Helix Proxy streams the cached version to the player, as illustrated in the next figure.

### On-Demand Clip Streamed from the Cache



### Notes About Content Caching

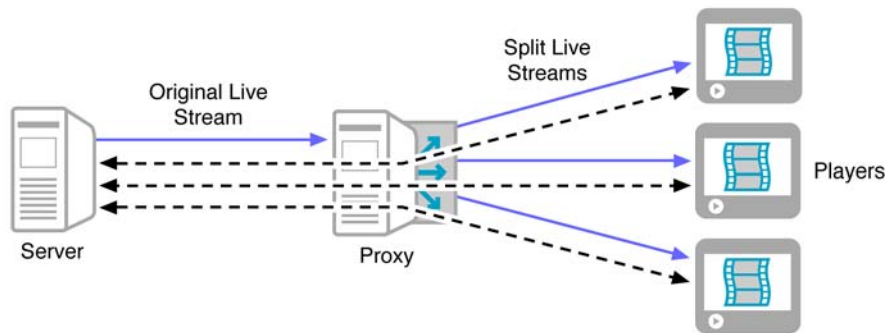
Note the following issues concerning content caching:

- On-demand material streamed by RealSystem Server 7.0 or later can be cached. Content streamed by earlier servers is delivered by using the pass-through method.
- If a clip contain multiple streams encoded at different bit rates, such as a SureStream clip, Helix Proxy preserves bandwidth by requesting and caching only the stream appropriate for the player's connection speed. If another media player requests the same clip at a different connection speed, Helix Proxy requests and caches the additional stream.
- Helix Proxy attains approval from Helix Server for each request before streaming a clip from its cache.
- If an origin Helix Server has been configured to prevent caching, Helix Proxy uses the pass-through feature to deliver content to media players, without caching the media.
- Should the media in the cache become impaired in some way, the stream halts and media players receive an error message.
- If the accounting connection between the media player and the origin Helix Server cannot be opened or is interrupted, Helix Proxy terminates the stream, sending the media player an error message.
- When the cache reaches its maximum size and new material needs to be cached, Helix Proxy deletes the cached files that have been requested least often.
- As explained in "Modifying the Media Cache" on page 43, you can modify the size and location of the cache, or even turn caching off.

## Pull-Splitting

Pull-splitting conserves bandwidth for live broadcasts by replicating a single, live stream from an origin Helix Server. The first time a media player requests a live stream, Helix Proxy opens its standard accounting connection with the origin server, then replicates the live stream to the media player. Subsequent media players that request the same live stream receive it directly from Helix Proxy.

### Pull-Splitting for Live Streams



### Notes About Pull-Splitting

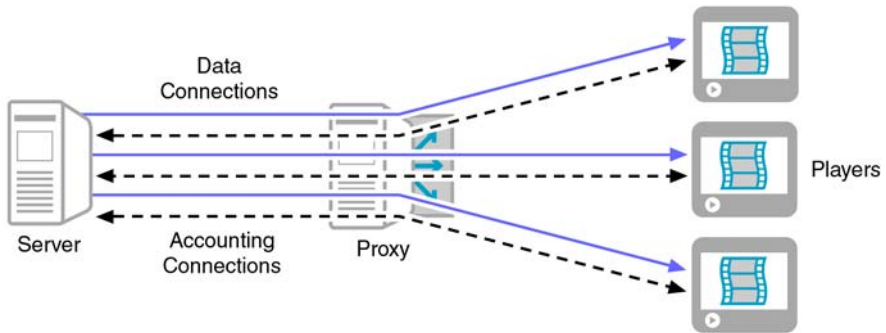
Note the following issues concerning pull-splitting:

- Even though it can split a single data stream to multiple players, Helix Proxy always opens a separate control channel to the origin Helix Server for each media player.
- The replicated, live stream from Helix Server to Helix Proxy uses UDP by default, or TCP if UDP is not supported.
- As explained in “Changing Pull-Splitting” on page 44, you can change pull-splitting operation, such as using only TCP for split streams, or turning pull-splitting off entirely.
- As described in Chapter 7, Helix Proxy can deliver a live, pull-split stream by multicasting it on a multicast-enabled network.
- Helix Proxy does **not** currently support the latency reduction feature for live broadcasts introduced with Helix Server Version 11. If Helix Proxy splits a live, low-latency stream to multiple RealPlayers, it buffers the stream, introducing broadcast latency.

## Pass-Through Delivery

Pass-through delivery is the simplest method of Helix Proxy operation. In addition to the standard accounting connection opened to Helix Server, Helix Proxy opens a data channel for each media player. In short, all content data comes from the server and passes through the proxy, which results in no bandwidth conservation.

### Pass-Through Mode for Live and On-Demand Streams



Pass-through delivery occurs if the origin Helix Server does not allow pull splitting or caching for requested clips or broadcasts. (Caching is not advisable in all cases, such as with frequently-updated material like advertisements.) A Helix Proxy administrator has no control over streaming splitting and caching directives set on Helix Server.

**Tip:** Although pass-through does not conserve bandwidth, Helix Proxy still acts as a portal for all streaming media traffic, enabling better monitoring and security.

## Additional Features

The following sections describe features that make it easy to configure, administer, and maintain Helix Proxy.

### Helix Administrator

Helix Administrator is a secure, HTML-based interface for customizing Helix Proxy features. You can access Helix Administrator with a Java-enabled browser anywhere on your network. Refer to “Using Helix Administrator” on page 37 for more information.

## XML-Based Configuration File

Changes you make using Helix Administrator are stored in a configuration file that Helix Proxy reads at start-up. This text file is based on XML (Extensible Markup Language) that you can edit directly. Because the file structure is complex, however, Helix Administrator is the recommended tool for making changes. Appendix A discusses the configuration file.

## Bandwidth Restriction

The section “Controlling Bandwidth” on page 52 explains how to restrict the number of requests or amount of bandwidth that Helix Proxy uses. Media players that attempt to contact Helix Server after Helix Proxy’s limits have been reached receive an error message.

## Proxy Routing

For organizations that use strict rules to regulate network traffic, proxy routing allows for further control of streaming media traffic. With this feature, which Chapter 6 explains, you can configure Helix Proxy to direct media player requests to another Helix Proxy, thereby conserving bandwidth across a wide area network.

## Proxy Redundancy

As explained in “Setting Up Proxy Redundancy” on page 54, you can instruct RealPlayer to contact an alternate proxy if the stream it receives from its current Helix Proxy terminates abnormally. This allows you to reconnect users to back-up proxies automatically in case of a network outage.

## Automatic Bandwidth Detection for Desktop RealPlayer

Helix Proxy uses a method of bandwidth detection that allows the desktop version of RealPlayer to receive the optimal stream when connected to different networks. This eliminates the need for the user to change the bandwidth configuration manually. For example, a RealPlayer on a laptop computer may automatically receive a 150 Kbps LAN stream when the computer is on an office network, a 512 Kbps stream when plugged into the user’s home DSL line, and a 34 Kbps modem stream when connecting through a dial-up modem while the user travels.

Note the following about automatic bandwidth detection:

- This feature works only with RealPlayer 11 and later. Earlier versions of RealPlayer prompt the user to set the optimal bandwidth choice when they detect a change in the network configuration. Other media players are not supported.
- Automatic bandwidth detection works only when RealPlayer uses the default RTSP control protocol and RDT packet format. It does not work with RTSP using the standards-based RTP packet format.

**For More Information:** See “Packet Formats” on page 122.

- Automatic bandwidth detection is compatible with all methods of launching RealPlayer. This includes Ram files, SDP files, and the Ramgen utility.
- If RealPlayer reconnects to a different Helix Proxy as described in “Setting Up Proxy Redundancy” on page 54, the second Helix Proxy performs a test to determine the optimal streaming bandwidth.
- In pass-through mode, described in “Pass-Through Delivery” on page 22, the origin Helix Server performs the bandwidth test rather than Helix Proxy.
- Automatic bandwidth detection is turned on by default. You can turn it off through the SendBWPackets variable in the Helix Proxy configuration file.

**For More Information:** Refer to *Helix Proxy Configuration and Registry Reference*.

## Access Control

Chapter 10 explains how to associate connection rules based on IP addresses. These rules can allow or deny connections to specific protocol ports.

## Authentication

Authentication, which Chapter 11 covers, verifies the identity of a user requesting streaming media. This verification can take the form of asking for a user name and password. Or, it can be entirely hidden from the viewer.

## Real-Time Monitoring

Chapter 14 explains the Proxy Monitor, which dynamically displays the status of your Helix Proxy.

## Activity and Error Logs

Helix Proxy records information about all streams it has served in its access log. It records operational errors in its error logs. Access logs include the address of the origin Helix Server, as well as the Helix Proxy delivery mode, such as pull-splitting, or caching. Chapter 12 explains the standard access and error logs. Chapter 13 explains how to customize reports of access and error activity.

## SNMP

Using Simple Network Monitoring Protocol (SNMP) version 1, 2c, or 3, you can monitor Helix Proxy from any SNMP-compliant management system. The SNMP feature allows you to monitor Helix Proxy performance and update proxy configuration remotely. It includes a master agent that acts as an intermediary between Helix Proxy and the management system. The agent can run as an independent application or a Windows service.

**For More Information:** Chapter 15 explains how to configure the SNMP feature.

## Administering a Server and Proxy Together

If you administer both Helix Proxy and Helix Server, keep in mind the following similarities and differences between the two products:

- Configuration files

The structure of the configuration files is the same on both products. However, certain configuration sections are unique to each product. You therefore cannot use a configuration file for Helix Proxy with Helix Server, and vice versa.

- Access logs

Both products have similar access logs. Helix Proxy's access log contains additional information appended to the end of each record, though. The log file for the origin Helix Server indicates whether a request was made

through Helix Proxy or directly by a media player. Both log files contain information about quality of service for the streams they deliver.

- Pull-splitting

Helix Proxy's pull-splitting method is nearly identical to that used by Helix Server. However, Helix Proxy does not need to include the transmitting Helix Server in its request URL.

- Multicasting

Helix Proxy uses only back-channel multicasting. It does not include the scalable multicasting feature available on Helix Server. Furthermore, Helix Proxy can multicast only the incoming streams that are enabled for pull-splitting on the origin Helix Server transmitter.

- Authentication

Like Helix Server, Helix Proxy authenticates users who access Helix Administrator. Unlike the server, Helix Proxy does not perform authentication on a per-clip basis. Instead, it allows or denies player access to specific servers by looking at the address of the origin Helix Server.

- Caching

Both Helix Server and Helix Proxy allow caching by default. Because Helix Server conserves bandwidth when caching is allowed, administrators are encouraged to leave all content cachable.



## CONFIGURATION

In this section, you learn how to install and start Helix Proxy, and use the Helix Administrator interface. Subsequent chapters explain how to configure the basic and advanced features of Helix Proxy.



## INSTALLATION AND TESTING

This chapter explains how to install Helix Proxy. It also introduces you to Helix Administrator, the Web-based tool for configuring Helix Proxy. As soon as you start Helix Proxy, it is ready to stream media, and the last section walks you through processes for configuring a media player to use your Helix Proxy.

### Understanding Installation Issues

Before you install Helix Proxy, you need to make basic set-up and deployment decisions, as described in the following sections.

#### Firewalls and Helix Proxy

You need to choose where to place Helix Proxy in relation to firewalls—either your firewall or an outside organization’s firewall—for optimal communication. Chapter 9 explains general issues involving firewalls. If your organization has a firewall, and you are not sure of its impact on Helix Proxy communication, be sure to read “Firewall Configurations” on page 125.

**Tip:** If you have questions about which ports are available on your network to allow traffic through a firewall, consult with your firewall administrator.

#### Web Servers and Helix Proxy

RealNetworks suggests that you do not install Helix Proxy on the same physical machine that runs your Web server. This eliminates conflicts over ports, and helps to balance loads so that Helix Proxy is not affected by heavy Web server use, and vice versa.

## Installing Helix Gateway on a Single Machine

RealNetworks recommends that you do **not** install both of the Helix Gateway applications (Helix Server and Helix Proxy) on the same machine. This prevents the applications from competing for processor power and memory. It is possible to install both applications on the same machine, however, if only one machine is available and you have low-volume streaming needs.

**For More Information:** Refer to *Helix Server and Helix Proxy Troubleshooting Guide* for instructions on installing and starting both Helix Server and Helix Proxy on a single machine.

## Installing Helix Proxy

To install Helix Proxy, you need a binary installation file and a license file that enables the Helix Proxy features. License files are delivered by e-mail after you receive your Helix Proxy installer and generate your machine ID, as described in Step 3 below.

**Tip:** To install Helix Proxy as a Windows Service, you must have administrative access.

**Warning!** If you are installing both Helix Server and Helix Proxy on the same machine, refer to the instructions in *Helix Server and Helix Proxy Troubleshooting Guide*. Otherwise, the applications will not start up properly.

► To install Helix Proxy:

1. Launch the binary setup file you downloaded. If you have a Helix Proxy installation CD, open the folder named for the operating system you are using, and execute the setup file.
2. Read the installation recommendations and press **Next**.
3. Click **Browse** to navigate to the license file you received from RealNetworks. Then press **Next**.

**Tip:** If you have not received your license file, copy the 32-character machine ID shown by the installer. Then, follow the instructions in the e-mail sent to you when you downloaded the Helix Proxy installer. This generates a license file valid for the machine on which the installer is running.

**Note:** The installation process copies the specified license file to the License subdirectory under the main Helix Proxy directory. On startup, Helix Proxy reads this local copy of the license. The original license file is no longer required, but should be stored securely in case re-installation is required.

**For More Information:** If you have problems generating the machine ID or receiving a license, contact RealNetworks Technical Support at <http://service.real.com>.

4. Read the end-user license agreement, signifying your agreement to its terms and conditions by clicking **Accept**.
5. Enter a path where you want to install Helix Proxy. On Windows, you may choose to accept the default path, or specify a different path.

**Note:** Examples in this guide assume that you've chosen the default path.

6. Enter a user name and password, and then confirm your password by entering it again. Your user name and password are required to access various Helix Proxy features, such as Helix Administrator.

**Tip:** Choose a password that is difficult to guess, and that includes both letters and numbers. The password is case-sensitive.

7. In the next set of screens, you define ports that Helix Proxy uses for the RTSP, HTTP, and MMS protocols, as well as the port used by Helix Administrator. RealNetworks recommends accepting the default ports, unless those port values will cause conflicts with other applications. Note the following:

- You can change the port settings after installation, as described in “Defining Communications Ports” on page 45.
- Although Helix Proxy does not use HTTP to deliver media, it reserves port 8080 for HTTP on startup. This generally avoids conflicts with Web server software that may be installed on the same machine. (A Web server typically uses port 80 as its default HTTP port.) If port 8080 is in use on the Helix Proxy machine, choose a different HTTP port when installing Helix Proxy.

- On UNIX, the default value for the RTSP port is lower than 1024, meaning that you have to log in as root to start Helix Proxy if you accept the default value.
  - You need the Admin port number to connect to Helix Administrator from a Web browser. As a security feature, the installer randomly generates this port number. RealNetworks recommends that you accept the default, but you can change the port value if you wish, or you know that the selected value will conflict with another port assignment. In either case, remember the port number, or record it in a secure location.
8. On Windows, the default installation sets up Helix Proxy as a service. This is recommended, but you can prevent this by clearing the **Run as NT Service** box.

**For More Information:** If you choose, you can later set up Helix Proxy to run as a service, as described in *Helix Server and Helix Proxy Troubleshooting Guide*.

This installer page also presents the option to **Install SNMP Master as an NT Service**. If you check this box, the Simple Network Monitoring Protocol master agent is installed as a service. This optional feature is significant only if you have licensed the SNMP feature, which Chapter 15 explains.

9. In the final confirmation screen, review the installation information. Click **Finish** to complete the installation process.

## Upgrading in a Different Directory

If you are upgrading, and you install Helix Proxy in a path that differs from that of your previous proxy installation, you need to move some of your existing files from the previous installation directory to the new directory after the installation. Optionally, you'll need to move files in the Logs directory. If you are using authentication, you'll also need to move the files described in Appendix C.

## Reinstalling Helix Proxy in the Same Directory

Reinstallation is generally not necessary, but if needed, you can reinstall Helix Proxy by repeating the installation procedure described in "Installing Helix

Proxy” on page 30. A reinstallation does not affect the proxy cache, but it resets your Helix Proxy configuration values to their defaults. If you tailored your system configuration after the initial installation, the following tips allow you to retain your data and make your reinstallation process smoother:

- Back up the configuration file (`rmproxy.cfg`) and SNMP configuration file (`master.cfg`) to preserve the configuration information. After the reinstallation, replace the files created by the installer with your backups.
- Back up any authentication databases (`adm_b_db`, `con_r_db`, and so on) that you’ve revised or added. This step is necessary only if you’ve added more users and passwords for authentication than those added during installation. Appendix C explains authentication databases.
- Note the value of the Admin port (**Proxy Setup>Ports**). If you bookmarked Helix Administrator in your browser, specify the same Admin port during the reinstallation to keep the bookmark functional.
- A reinstallation does not affect cache files, access logs, or error logs. It is therefore not necessary to back up these files before reinstallation. These files reside in the Cache and Logs subdirectories of the main installation directory.

## Running Helix Proxy

This section describes how to start and stop Helix Proxy on Windows and UNIX. Helix Proxy requires a minimum of 256 MB of application memory.

**For More Information:** For information about recommended operating system modifications, application memory usage, and system start-up parameters, refer to the chapters on operating system modifications and command options in *Helix Server and Helix Proxy Troubleshooting Guide*.

## Starting Helix Proxy

When you start Helix Proxy manually, you can select which configuration file you want to use. As described in “Restarting Helix Proxy” on page 41, you can use Helix Administrator to restart Helix Proxy following a configuration change.

**Tip:** The standard configuration file does not contain relative paths to components. However, a configuration file may be

modified to include relative paths. In this case, you must start Helix Proxy from the directory that holds the configuration file to resolve the relative paths correctly. For this reason, the following sections show how to start Helix Server from its main installation directory, which contains the standard configuration file.

## Starting on Windows

The following sections explain how to start Helix Proxy as a Windows service, from the **Start** menu, or from the Windows command line.

### Changing the Windows Service Startup Parameters

In its default Windows installation, Helix Proxy is set up as a service named Helix Proxy. In this case, Helix Proxy always runs in the background, and you do not need to start it. You may wish to increase its memory maximum, however, as described in “Changing the Maximum Memory Usage of the Service” on page 35.

### Starting Up from the Start Menu or Desktop

From the **Start** menu, select **Programs>Helix Proxy>Helix Proxy**. Or, double-click the Helix Proxy icon on the Windows desktop. This starts Helix Proxy with its default configuration file, `rmproxy.cfg`, and a memory maximum of 256 MB.

To change the configuration file or to adjust the maximum memory usage, stop the proxy if it is running. Right-click the icon you use to start the proxy and select **Properties**. In the Target field, you can change `rmproxy.cfg` to the name of a new configuration file. To increase the memory usage, add `-m 512` to the end of the field as shown in the following example, then restart the proxy:

```
"C:\Program Files\Real\Helix Proxy\Bin\rmproxy.exe" "C:\Program Files\Real\Helix Proxy\rmproxy.cfg" -m 512
```

**Note:** You need to follow this procedure for each shortcut icon that you use to start Helix Proxy.

### Starting Up from the Command Line

From the **Start** menu, open the command prompt. Navigate to the Helix Proxy folder, and enter the following command to start Helix Proxy with its default configuration file and standard memory use (256 MB):

```
Bin\rmproxy rmproxy.cfg
```

Optionally, you can use a different configuration file, as well as change the maximum memory allotment by including the `-m` parameter. After the `-m` parameter, specify the amount of memory in Megabytes (must be 256 MB or greater). The following example allows Helix Proxy to use up to 512 Megabytes of memory:

```
Bin\rmproxy rmproxy.cfg -m 512
```

### Changing the Maximum Memory Usage of the Service

By default, the Helix Proxy service uses a maximum of 256 Megabytes of memory. Follow the next procedure to change this amount for a running service.

► **To change Helix Proxy memory usage:**

1. Choose **Start>Settings>Control Panel**.
2. Double-click **Administrative Tools**
3. Double-click **Services**.
4. Locate Helix Proxy in the list, highlight it, right-click, and choose **Stop**.

**Note:** Your service name may be different if you set up the service after installation, as described in *Helix Server and Helix Proxy Troubleshooting Guide*.

5. Highlight Helix Proxy on the list, right-click, and choose **Properties**.
6. Navigate to the **General** tab.
7. In the **Start parameters** field, enter the `-m` parameter followed by the maximum amount of memory in Megabytes. For example:  
`-m 512`
8. Click **OK**.
9. Highlight Helix Proxy on the list, right-click, and choose **Start**.

### Starting on UNIX

If you performed a default installation of Helix Proxy, the RTSP port is set lower than 1024, requiring the user who starts Helix Proxy to log in as root. If you do not want Helix Proxy to inherit root privileges, you can switch Helix Proxy to another user and group name immediately after it starts up. For instructions, refer to “Setting UNIX User and Group Names” on page 53.

You can start Helix Proxy as an application or as a background process. The following procedure uses the default configuration file (`rmproxy.cfg`), but you can specify a different file.

► **To start Helix Proxy on UNIX:**

1. Start any command shell.
2. Navigate to the main Helix Proxy installation directory.
3. Choose one of the following options:
  - a. Start Helix Proxy in the background with the following command:  
`Bin/rmproxy rmproxy.cfg &`
  - b. Start Helix Proxy as an application:  
`Bin/rmproxy rmproxy.cfg`
  - c. Optionally, you can adjust the amount of memory that Helix Proxy can use from the default value of 256 MB. Do this by including the `-m` parameter, where the number after `-m` specifies the amount of memory in Megabytes (must be 256 MB or greater). The following example starts Helix Proxy as an application:

```
Bin/rmproxy rmproxy.cfg -m 512
```

The next example starts Helix Proxy as a background process:

```
Bin/rmproxy rmproxy.cfg -m 512 &
```

**Tip:** If the Helix Proxy machine is dedicated to running Helix Proxy, RealNetworks recommends that you allocate 75 percent of the available system memory for Helix Proxy's use.

#### Process ID (PID)

Helix Proxy creates a text file that records the current value of the process ID of the parent Helix Proxy process, `rmproxy`. The file is stored in the directory indicated by the `PidPath` variable, and is named `rmproxy.pid` at installation. If `PidPath` is omitted from the configuration file, Helix Proxy stores the information in the directory specified by the `LogPath` variable.

## Stopping Helix Proxy

It's generally not necessary to stop Helix Proxy when it's running. If you make configuration changes that require a restart, you can restart through Helix Administrator, as described in "Restarting Helix Proxy" on page 41.

**For More Information:** The section “Implementing Delayed Shutdown” on page 50 explains how to allow media players to report playback statistics before the shutdown commences.

### Shutting Down on Windows

If Helix Proxy was started as a Windows service, stop it through the **Services** control panel. Give the **Start>Settings>Control Panel>Administrative Tools** command and double-click **Services**. Locate Helix Proxy on the list (your service name may be different), highlight it, and click **Stop**.

If you started Helix Proxy manually, switch to the command window and press **Ctrl+c**. You can also use the Task Manager (**Ctrl+Shift+Esc**) to end the Helix Proxy task.

### Shutting Down on UNIX

To stop Helix Proxy on UNIX, obtain the parent process identification number, and then issue the kill command with that process number. The process ID is stored in the `rmproxy.pid` file, which is usually kept in the `Logs` directory. (The `PIDPath` variable in the configuration file specifies this location.) You can perform both actions with one command. From the command line, navigate to the directory that contains the Helix Proxy PID file, and type the following, where *pidfile* is the name of the PID file:

```
kill `cat pidfile`
```

## Using Helix Administrator

Helix Administrator is Helix Proxy’s HTML-based interface. It allows you to modify and manage Helix Proxy from anywhere on your network using a Web browser.

**Tip:** In this guide, “Helix Administrator” with an uppercase “A” refers to this HTML-based tool, whereas “Helix administrator” with a lowercase “a” refers to the person who configures and runs Helix Proxy.

**Note:** You can configure most Helix Proxy features by using Helix Administrator. However, some advanced features require manual editing of the Helix Proxy configuration file, as described in *Helix Proxy Configuration and Registry Reference*. Integrating Helix Proxy with other systems may also require

manual configuration changes. For this information, refer to your systems documentation.

## Starting Helix Administrator

To start Helix Administrator, you need to know the port number it uses, as well as the user name and password selected during Helix Proxy installation. The password selected during installation is stored in the `MonitorPassword` variable of the configuration file. For background on the configuration file, see Appendix A.

► **To start Helix Administrator:**

1. Start Helix Proxy if it is not already running. See the section “Starting Helix Proxy” on page 33 for startup instructions.

2. Click the browser shortcut added to the desktop by the Helix Proxy installer, or open the following location in your Web browser:

`http://address:AdminPort/admin/index.html`

If your browser is on the same machine as Helix Server, you can typically use the localhost address:

`http://localhost:AdminPort/admin/index.html`

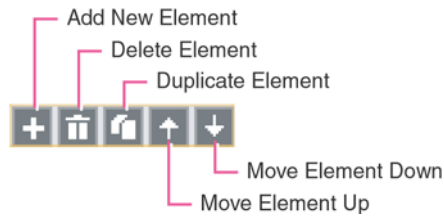
3. Enter the user name and password chosen during installation. The password is case-sensitive.
4. Click **OK** to start Helix Administrator.

**Tip:** You can create additional user names and passwords to let other people access Helix Administrator. For more information, see “Administrator Authentication” on page 140.

## Navigating the Interface

Helix Administrator consists of HTML pages you use to configure Helix Proxy. The left frame groups features into functional areas, as described below. Pages that display in the right pane typically consist of forms that include fields and pull-down lists. In pages that define multiple elements, you can use the control icons shown in the following illustration.

### Helix Administrator Controls



When you change configuration information on a Helix Administrator page, click **Apply** at the bottom of the page to save the changes. An arrow appears next to the **Apply** button and the page title tab to indicate that changes require saving. A confirmation dialog appears when you click **Apply**. Note that Helix Administrator discards changes if you navigate to a different page before clicking **Apply**. As well, clicking **Reset** returns the current page to its stored values.

**Tip:** If you are familiar with previous versions of RealSystem Administrator, note that you no longer have to click an **Edit** button to update an element definition. You simply enter the element information in the fields, and click **Apply** at the bottom of the page when you are finished.

### Helix Administrator Sections

Helix Administrator's left-hand navigation pane groups Helix Proxy features under functional areas such as **Security**. Click the name of a functional area to expand or collapse the list of features it contains. The following tables summarize all features, and point you to the sections of this guide that explain each feature.

#### Proxy Setup

The proxy setup features let you configure the basic functions of Helix Proxy. Many of these features are preconfigured at installation.

**Proxy Setup Features**

Feature	Function	Reference
Ports	Define ports for communications protocols.	page 45
IP Binding	Select IP addresses Helix Proxy uses.	page 48

(Table Page 1 of 2)

**Proxy Setup Features (continued)**

Feature	Function	Reference
Bandwidth Management	Limit the amount of bandwidth Helix Proxy uses.	page 52
Proxy Routing	Set up proxy chains to other Helix Proxies.	page 97
Redundant Proxies	Define failover proxies for streaming delivery.	page 54
Cache	Set limits to caching of on-demand content, and rename the cache directory.	page 43
Delayed Shutdown	Add time to close connections on a shutdown.	page 50
Splitting	Modify the protocol to use with pull-splitting and determine whether to allow packet resends.	page 44
Multicasting	Configure back-channel multicasting.	page 103
Media Delivery	Set up server-side rate control.	page 61

(Table Page 2 of 2)

**Transport Settings**

The transport settings affect how Helix Proxy delivers streams.

**Transport Settings**

Feature	Function	Reference
Differentiated Services	Mark IPv4 packets for specific treatment.	page 56

**Security**

The security features allow you to limit connections to Helix Proxy by media player IP address, or by user name and password..

**Security Features**

Feature	Function	Reference
Access Control	Limit media player connections by IP address.	page 133
User Databases	Select authentication databases.	page 147
Realms	Create authentication realms.	page 149
Authentication	Enable authentication and choose no-authenticate sites.	page 143

## Logging and Monitoring

The logging and monitoring features let you view current Helix Proxy activity, as well as review past activity.

### Logging and Monitoring Features

Feature	Function	Reference
Proxy Monitor	Display statuses of current connections.	page 199
Basic Logging	Compile user and error statistics.	page 155
Advanced Logging	Create templates for reports.	page 183
SNMP	Implement Simple Network Monitoring Protocol	page 203

## Restarting Helix Proxy

Some configuration changes you make to Helix Proxy require a restart, which breaks open connections for live events or clips streamed on demand. It's best, therefore, to make these changes during periods of low use. The Helix Administrator interface indicates feature changes that require a Helix Proxy restart. It also prompts you when a change requires a proxy restart when you click **Apply**. Click the **Restart Server** button to restart Helix Proxy.

**For More Information:** The section “Implementing Delayed Shutdown” on page 50 explains how to allow media players to report playback statistics before the restart commences.

### Queuing Changes for a Later Restart

It is not necessary to restart Helix Proxy immediately after you make a configuration change. In this case, the **Pending Changes** flag appears in the upper-right corner of Helix Administrator. This flag reminds you that all pending changes will go into effect the next time Helix Proxy is started.

## License File Information

At least one valid, text-based license file must reside in the License subdirectory of Helix Proxy's installation directory. This file is in an XML format that you can read with a text editor. You can also display the license file through Helix Administrator by clicking **About**. You generally do not need to do anything with the license file, as long as Helix Proxy reads it correctly on startup.

**Tip:** If you have multiple license files, Helix Administrator shows the values for all of the licenses at once. In this case, you need to read each file individually and calculate additive features, such as the total number of licensed streams.

**Warning!** Making any changes to a license file invalidates the license and prevents Helix Proxy from starting. To resolve this, contact RealNetworks for a valid license file.

## Testing Helix Proxy

In this section, you'll use RealPlayer to play content directly from a Helix Server, then configure it to use Helix Proxy. You'll need to start Helix Proxy as described in "Running Helix Proxy" on page 33, and have Helix Administrator running, as described in "Running Helix Proxy" on page 33. Additionally, you'll need RealPlayer, which is available for download from <http://www.real.com>.

► **To test Helix Proxy:**

1. Play sample content from a Helix Server. You can use a clip streamed from your own Helix Server, or a clip available through one of RealPlayer's channels.
2. Use the Proxy Monitor described in Chapter 14 to verify that Helix Proxy isn't in use. All numbers in the columns should show a value of zero.
3. Configure RealPlayer to use your Helix Proxy as described in "Configuring RealPlayers Manually" on page 111.
4. Use RealPlayer to request the same content played in the first step.
5. Look again at the Proxy Monitor. The numbers will be different, indicating that Helix Proxy is proxying the stream to your RealPlayer.

## PROXY SETUP

This chapter describes basic Helix Proxy setup. These functions include specifying ports, binding to IP addresses, and enabling pull-splitting. You may not need to change any of these settings depending on your system's configuration and the values you chose during installation.

## Modifying the Media Cache

The media cache, which is described in the section “Content Caching” on page 19, is enabled by default. You do not need to change any settings on Helix Proxy to use this feature. However, you can modify the cache size, change the directory used for the cache, or even turn off caching.

► To modify the cache:

1. In Helix Administrator, click **Proxy Setup>Cache**.
2. If you want to disable caching, choosing **No** from the **Enable Caching** pull-down list. Disabling caching eliminates the bandwidth conservation benefits you receive from Helix Proxy when streaming clips on-demand.
3. To change the size of the cache, set the cache set in Megabytes in the **Maximum Cache Size** field. The default value is 1000 Megabytes (approximately 1 Gigabyte). The minimum value you can use is 11 Megabytes.

**Tip:** To increase the bandwidth conservation benefits of using Helix Proxy, make the cache as large as possible.

4. In the **Cache Directory** field, define the full or relative path to the directory used as the cache. Relative paths are relative to the Helix Proxy installation directory. The default directory is `Cache`, which is located in the Helix Proxy installation directory.

**Tip:** RealNetworks recommends that you always cache files on a hard disk on the Helix Proxy machine. Using a directory on a network drive may slow media delivery.

**Note:** If you change the location of the cache, do **not** copy the existing cache files to the new location. Helix Proxy will rebuild the cache in the new location through its normal operation. After you have verified that the new cache is working properly, you can delete the obsolete cache files.

5. Click **Apply**.
6. Restart Helix Proxy.

## Changing Pull-Splitting

Pull-splitting, which is described in the section “Pull-Splitting” on page 21, is enabled by default. When a media player requests a live stream, Helix Proxy acquires the stream from the origin Helix Server through the highly efficient pull-splitting method when possible. If necessary, you can turn off pull-splitting for Helix Proxy, or set the transport protocol used for the broadcast stream between server and proxy.

► **To change the pull-splitting settings:**

1. Click **Proxy Setup>Splitting** in Helix Administrator.
2. To turn off pull-splitting, choose **No** in the **Attempt to Split All Live Broadcasts** pull-down list.
3. The **Live Splitting Transport** list defines the transport-layer protocol to use with pull-splitting. You have these choices:
  - Always use TCP
  - Always use UDP
  - Auto Negotiate

Auto-negotiation is the default. In this case, Helix Proxy expects to receive origin streams over UDP. If UDP is unavailable, stream delivery uses TCP.

**Tip:** UDP delivery is preferable. Origin streams using TCP may result in media player rebuffering or greater start-up latency when a media player connects. However, if you set UDP-only

delivery, ensure that any firewalls between the server and proxy do not block UDP delivery.

4. The **Enable Resends** pull-down list is set to Yes by default. This setting allows data packets to be resent from the origin server. Setting this to No can conserve bandwidth but result in lost packets that lower the quality of service for media players.
5. Click **Apply**.

## Pull-Splitting with RealSystem Server Version 8

If you use RealSystem Server version 8 as the origin server for Helix Proxy pull-splitting, you need to modify the value of a variable in the Helix Proxy configuration file.

► **To enable pull-splitting with RealSystem Server 8:**

1. Open the Helix Proxy configuration file in a text, XML, or HTML editor. The default configuration file is `rmproxy.cfg`, and is located in the main Helix Proxy installation directory.

**For More Information:** See Appendix A for information about the configuration file syntax.

2. Change the value for the variable `Splitter_DoubleURLEnable` from 0 to 1. The variable should look like the following:
 

```
<Var Splitter_DoubleURLEnable="1"/>
```
3. Save and close the configuration file.
4. Restart Helix Proxy.

## Defining Communications Ports

After you install Helix Proxy, you can change the ports used for protocols such as RTSP and MMS, as well as for features such as Helix Administrator. RealNetworks recommends that, whenever possible, you use the default communications ports, which are “well-known” ports that Web browsers and media players use by default when contacting Helix Proxy. The following table

lists the default ports for the protocols that Helix Proxy uses. Not all port numbers are editable through Helix Administrator.

#### Recommended Port Numbers

Protocol or Feature	Default Port	Purpose
RTSP	554	RTSP-based communication between Helix Proxy, RealPlayer, and QuickTime Player.
MMS	1755	MMS-based communication between Helix Proxy and Windows Media Player.
HTTP	8080	HTTP-based communication.
Admin	(random)	Communication with Helix Administrator. The value is randomly generated during installation, and is required in the browser URL when connecting to Helix Administrator.
Monitor	9090	Communication with the Proxy Monitor. For more information on this feature, see Chapter 14.

## Shared UDP Port Ranges

Helix Proxy establishes a UDP channel for media player packet acknowledgements and packet resend requests. Certain network and firewall configurations might prevent UDP data from being sent between a media player and Helix Proxy, or between Helix Proxy and Helix Server.

Although network prohibition of UDP does not halt RTSP traffic and media player playback, it does restrict the ability of the proxy and server to respond to packet loss, and can degrade the quality of service. Thus, Helix Proxy and Helix Server allow all UDP client-originated traffic to be multiplexed through a limited, shared UDP port range, if this configuration is required by your firewall.

**For More Information:** See Chapter 9 for background information on firewalls, ports, and protocols.

## Changing Streaming Media Ports

You can easily change ports through Helix Administrator. This requires a Helix Proxy restart.

► **To change Helix Proxy port settings:**

1. Click **Proxy Setup>Ports**.

2. Change values by entering new numbers for various ports. When changing port numbers, keep the following points in mind:
  - If you change the port numbers for RTSP or MMS, you may need to reconfigure media players to contact Helix Proxy on the new ports. Refer to Chapter 8 for more information.
  - If you change the Admin port, you need to log into Helix Administrator again with the new port value after you restart Helix Proxy.
3. Initially blank, the **UDP Resend Port Range** fields instruct Helix Proxy to use the specified range of UDP ports for media player replies. Enter a minimum range of two ports for each CPU on the Helix Proxy machine. The first port value used in this variable must always be an even number.
4. Click **Apply**. If you receive an error message that the port is used for UDP communications, choose another port, or restrict the UDP range.
5. Restart Helix Proxy.

## Changing the HTTP Port

Although Helix Proxy does not use HTTP to deliver media, it reserves an HTTP port on startup. The default port value is 8080. If you need to change the HTTP port value, you can edit the Helix Proxy configuration file.

► **To modify the value of HTTP Port:**

1. Open the configuration file in a text, XML, or HTML editor. The default configuration file is `rmproxy.cfg`, and is located in the main Helix Proxy installation directory.

**For More Information:** See Appendix A for information about the configuration file syntax.

2. Find the variable `HTTPPort` and change its value to a port that is not in use.
3. Save and close `rmproxy.cfg`.
4. Restart Helix Proxy.

## Binding To An IP Address

When Helix Proxy starts, it uses the IP address assigned to the first network interface it finds on the computer—network interface 0. In a computer with multiple network interfaces—often referred to as a *multi-homed* machine—you can configure Helix Proxy always to use specific IP addresses. Through this feature, you can select individual IP addresses to use, whether IPv4 or IPv6, or you can bind to all IP addresses on the machine.

**For More Information:** For more about IPv6 addresses, refer to “IP Version 6” on page 16.

### Using Localhost

By default, Helix Proxy binds to the IPv4 or IPv6 *localhost* address (also called the *loopback* address), which enables a simulated network connection from Helix Proxy to a client installed on the same computer. When using this address, which is useful for testing, no information is sent over the network, but it appears as if the connection came from the network. You can express the IPv4 localhost address in dotted decimal form as 127.0.0.1. For IPv6, the localhost address is 0:0:0:0:0:0:0:1, which can be shortened to ::1.

### Capturing All Addresses

You can use the IP binding feature to capture all addresses for Helix Proxy’s use. To do this, specify one of the following options and delete all other address entries.

**Syntax for Capturing All IP Addresses**

Binding	Option 1
All IPv4 addresses	0.0.0.0
All IPv6 addresses	::
All IPv4 and IPv6 addresses	*

**Tip:** Helix Proxy automatically binds to all addresses and to localhost. For most installations, RealNetworks recommends binding to all addresses.

## Modifying IP Addresses

You bind Helix Proxy to IP addresses using Helix Administrator. You'll need to restart Helix Proxy after making these changes.

► **To reserve IP addresses for Helix Proxy:**

1. In Helix Administrator, click **Proxy Setup > IP Binding**.
2. Click the “+” icon and type the IP address that you want Helix Proxy to use into the **Edit IP Address** box.

**Warning!** Type the address carefully. If you type an IP address that does not exist on this computer, Helix Proxy will not be able to start.

**Note:** When you bind Helix Proxy to one or more specific IPv4 or IPv6 addresses, it also binds to the localhost address automatically. You can disable the automatic binding to localhost by specifying an IP address with the `--hbi` heartbeat check option when starting Helix Proxy. For more information, refer to *Helix Server and Helix Proxy Troubleshooting Guide*.

3. Repeat this procedure for each address on this machine that you want Helix Proxy to use.

**Warning!** If you use an option to capture all IP addresses of a certain type, as described in “Capturing All Addresses” on page 48, do not specify any other addresses of that type. For example, if you specify `0.0.0.0`, do not include any other IPv4 addresses. If you do, Helix Proxy will not be able to start up.

4. Click **Apply**.

**Note:** If a firewall is in use, you may need to configure it to allow traffic to pass on the addresses you added to the IP Binding list. See “Working with Multiple IP Addresses” on page 126 for information.

## Implementing Delayed Shutdown

The delayed shutdown feature lets you initiate a Helix Proxy shutdown or restart in a manner that allows media players to report playback statistics. This feature is useful for services in which it is necessary to know how much of a clip a media player received before the shutdown. If you do not implement this feature, Helix Proxy terminates all streams and shuts down immediately upon request, receiving no playback statistics from media players.

**For More Information:** The sections “Stopping Helix Proxy” on page 36 and “Restarting Helix Proxy” on page 41 explain how to initiate a shutdown or restart, respectively.

### Defining a Delayed Shutdown

The following procedure describes how to implement the delayed shutdown feature and define the shutdown intervals. This feature is not turned on by default.

► To define a delayed shutdown:

1. In Helix Administrator, click **Proxy Setup>Delayed Shutdown**.
2. For **Player Disconnect Interval**, enter the time in seconds that elapses between the shutdown request and cessation of all media streams. A value of 30, for example, gives media players 30 seconds of streaming time. This allows players nearing the end of a clip the opportunity to finish the session normally. A value of 0 stops streams immediately, but still allows media players time to report playback statistics.
3. The **Shutdown Proceed Time** sets the number of seconds during which Helix Proxy terminates active streams and receives statistics from media players before shutting down. A value of 30 is typically sufficient. You may want to set a higher value if your Helix Proxy regularly streams more than 1,000 simultaneous clips.

**Tip:** The shutdown delay is the sum of the two intervals. If you set 30 seconds for both the player disconnect interval and the shutdown proceed time, for instance, Helix Proxy shuts down 60 seconds after the request is given

4. If you select Yes in the **Log Player Termination Status** pull-down list, Helix Proxy records in its error log the number of successful and unsuccessful

media player terminations that occurred before the shutdown. It records the following statistics:

- Number of media players that ended their media sessions normally during the player disconnect interval.
- Total number of media players whose media sessions were terminated by the shutdown.
- Number of media players that successfully logged playback statistics in the access log file.
- Number of media players that ignored the termination request.

**Note:** The **Log Player Termination Status** setting affects only aggregate statistics for media player termination written to the error log. It does not affect the collection of playback statistics from each media player.

**For More Information:** See “Basic Error Log” on page 156 for a description of the error log.

5. If you set the **Allow New Client Connections During Shutdown** pull-down to No, Helix Proxy accepts no new media requests once the shutdown request is given. If you set this value to Yes, Helix Proxy accepts new requests until the player disconnect interval elapses. For example, if you define a long disconnect interval, such as 300 seconds, you may want to allow connections during shutdown. This lets media players connect and receive all or part of a clip until the player disconnect interval expires.

6. Click **Apply**.

**Tip:** If you want to implement this feature with the predefined values, simply click **Apply** on the shutdown page.

## Notes on Delayed Shutdown

Keep the following points in mind when implementing delayed shutdown:

- The delayed shutdown feature functions with RealPlayer, QuickTime Player, and Windows Media Player. Other media players may or may not respect the request to terminate playback and report playback statistics.
- On Linux and Sun Solaris, delayed shutdown occurs when you terminate Helix Proxy using killall or pkill, respectively. However, a kill -9 command

terminates Helix Proxy processes immediately without the shutdown delay.

- On Windows, delayed shutdown occurs when you terminate Helix Proxy using Ctrl+c or a Stop Service command. Stopping Helix Proxy through the Task Manager (Ctrl+Alt+Del) ends the processes immediately without the shutdown delay, however.
- Delayed shutdown is not initiated if Helix Proxy shuts down or restarts due to a hardware malfunction or a software error, such as running out of memory.

## Controlling Bandwidth

Helix Proxy provides several methods for managing bandwidth and media player connections on your network. By modifying the default settings of Helix Proxy, you can do the following:

- Limit the number of media players that can connect to Helix Proxy at one time.
- Restrict the bandwidth in use between Helix Proxy and media players.
- Restrict the bandwidth in use between Helix Proxy and other servers.
- Automatically disconnect an RTSP session that has timed out.

By default, Helix Proxy uses a value of 0 (zero) for these connection parameters, which represents the highest allowable values based on your Helix Proxy license agreement. Through Helix Administrator, you can set lower (but not higher) values for these connection parameters.

► To change connections parameters:

1. In Helix Administrator, click **Proxy Setup>Bandwidth Management**.
2. In the **Maximum Client Connections** box, type the number of media player connections, from 1 to 32767, that you want to allow simultaneously. This number can be less than or equal to the number of streams permitted by your license, which is summarized in the **Maximum Licensed Client Connections** field.

**Note:** If a media player requests a clip after the limit has been reached, Helix Proxy refuses the connection and sends the media player an error message.

3. The **Connection Timeout** field defines the number of seconds that can elapse without a client (either a media player or a proxy) sending any feedback to Helix Proxy. After this time expires, Helix Proxy queries the client for its status. If the client does not respond, Helix Proxy closes the connection.

**Note:** This timeout value applies only to RTSP-based connections.

4. In the **Maximum Proxy Bandwidth** box, enter the maximum number of kilobits per second (Kbps) that Helix Proxy can consume when streaming content to media players. For example, specify 1024 to limit the bandwidth to one Megabit.

**Note:** Helix Proxy can exceed the specified number. For example, if the limit is 100 Kbps, and one connection uses 80 Kbps, Helix Proxy will accept a subsequent connection of 40 Kbps even though the total bandwidth in use is 120 Kbps. However, no new media players will be permitted to connect after that.

5. The **Maximum Gateway Bandwidth** field sets the maximum bandwidth in Kilobits per second (Kbps) that Helix Proxy uses for connections to another Helix Proxy, Helix Server, or other Internet resources. Limiting gateway bandwidth can limit pass-through data connections, pull-splitting data connections, and initial cache requests. To limit the bandwidth to two Megabits, for example, specify 2048.

**Note:** As with downstream bandwidth, Helix Proxy can exceed the upstream bandwidth. For example, if you set a limit of 2048 Kbps, and Helix Proxy makes one connection that uses 1024 Kbps, it can make another connection of 1600 Kbps. It will make no further connections, however, until the bandwidth in use falls below the specified maximum.

6. Click **Apply**.

## Setting UNIX User and Group Names

By default, Helix Proxy on UNIX uses the user and group names of the person who starts it. After startup, though, it can immediately switch to a different

user and group setting. This lets you start Helix Proxy as root, so that it can capture port 554 for RTSP communications, then assume a different user and group identity. The user and group names must be predefined through the operating system, and must have write permission for Helix Proxy's Logs and Cache directories, as well as the Helix Proxy configuration file.

► **To change the group or user names:**

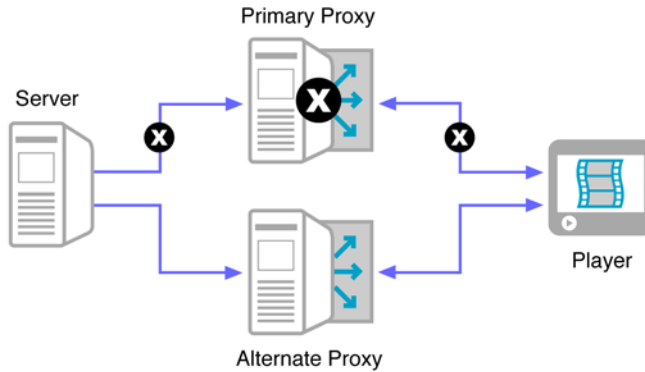
1. In Helix Administrator, click **Proxy Setup>User/Group Name**.
2. Type the user name or ID number in the **User Name or ID** box. The default is %-1, which means Helix Proxy uses the name of the user who logged in and started Helix Proxy.
3. Type the user name or ID number in the **Group Name or ID** box. The default is %-1, which means that Helix Proxy uses the group name of the user who logged in and started Helix Proxy.
4. Click **Apply**.

## Setting Up Proxy Redundancy

In general, if an RTSP connection between RealPlayer and Helix Proxy breaks, RealPlayer attempts to reconnect to the same Helix Proxy. However, you can identify any number of alternate proxies that RealPlayer uses instead to re-establish the stream for an on-demand clip or live broadcast. This redundancy improves quality of service if one Helix Proxy becomes inaccessible.

Proxy redundancy works only with RealPlayer and RTSP connections. When it sets up the RTSP control channel, Helix Proxy sends RealPlayer a list of alternate proxies to use should an unexpected disconnection occur. When RealPlayer recognizes the failure, it uses this list to connect to an alternate Helix Proxy. The following illustration depicts RealPlayer connecting to an alternate Helix Proxy after a primary proxy failure.

### Redundant Proxies



**Note:** The process of sending client requests made to one proxy through another proxy is called *proxy routing*, and is explained in Chapter 6. The redundant proxies and proxy routing features are distinct. The former protects against a failed proxy connection, whereas the latter manages proxy traffic flow.

### Requirements for Using Redundant Proxies

Using the redundant proxy feature requires that alternate proxies work the same as the primary Helix Proxy in the following respects:

- Alternates must have access to the same origin Helix Server as the primary Helix Proxy. This gives alternate proxies access to exactly the same content—whether live broadcasts or on-demand content—as the primary Helix Proxy.
- Alternates should have a similar configuration to the primary Helix Proxy. For example, access control rules and authentication databases of an alternate proxy should grant the same access as the primary proxy. If authentication is enforced, users must reverify their user name and password with the alternate proxy.

### Setting Up Redundant Proxies

Follow the procedure below to set up an alternate proxy for Helix Proxy. Each proxy can define any number of alternates. If you define multiple alternates, each RealPlayer randomly selects an alternate proxy from the list it receives. Therefore, the order in which you define alternate proxies does not matter.

**Tip:** Typically, you define a separate list of alternates on each Helix Proxy. Proxy A might designate Proxy B and Proxy C as its alternates, for example. Proxy B uses as its alternates Proxy A and Proxy C, whereas Proxy C designates Proxy A and Proxy B as its alternates.

► To define an alternate proxy:

1. In Helix Administrator, click **Proxy Setup>Redundant Proxies**.
2. In the **Alternate Proxies** box, click the “+” icon.
3. For the generic **Description** value, enter any description you like. This is for your use only, and does not affect how proxy redundancy operates.
4. In the **Host** box, enter the host name, IPv4 address, or IPv6 address of the alternate proxy.
5. For the **Port** value, specify the RTSP port used on the alternate proxy.
6. Click **Apply**.

## Configuring Differentiated Services

The differentiated services feature allows you to assign priorities to the IPv4 packets that carry streaming media data. These priority values can affect how the packets are forwarded through the network by routers that support differentiated services. For example, these values can cause routers to delay the forwarding of low-priority packets when network traffic is high. Using differentiated services can thereby increase a network’s efficiency by dividing packets into different classes that are assigned different delivery criteria.

**Note:** The differentiated services feature functions only with IP version 4 (IPv4). It does not support IP version 6 (IPv6) packets. The priority assignments are ignored if IPv6 is used.

**For More Information:** The Helix Proxy implementation of differentiated services complies with IETF standards described on the Web pages <http://www.ietf.org/rfc/rfc2474.txt> and <http://www.ietf.org/rfc/rfc2475.txt>.

## Network Requirements for Differentiated Services

Marking packets for differentiated services has an effect only on networks in which each node through which a packet passes is configured to read the IPv4 packet's differentiated services field (DSFIELD), and forward the packet based on the field's encoded criteria. This limits the benefits of differentiated services criteria to an intranet on which each router is enabled for these services, and the network administrator has defined rules for forwarding packets based on the DFIELD values.

**Note:** You can still stream media on networks that are not enabled for differentiated services, such as the Internet. However the priority values have no effect on packet forwarding, and each packet is treated the same.

## IP Header Bit Values

The settings of six bits within each IPv4 packet header define the differentiated services criteria.

### Precedence Bit Settings

Three bits within an IPv4 packet head define a precedence, which categorizes each packet. The following table lists the eight possible bit combinations that define each precedence. A precedence setting has no inherent meaning. Setting an IPv4 packet to the Priority precedence, for example, has significance only within a network that is configured to handle packets of this precedence in a specific way under certain network conditions. This packet handling can vary from network to network, and is wholly defined by each network administrator.

**Precedence Bit Settings**

Precedence	Value	Bit Setting
Routine	0	000
Priority	1	001
Immediate	2	010
Flash	3	011
Flash override	4	100
CRITIC/ECP	5	101

(Table Page 1 of 2)

**Precedence Bit Settings (continued)**

Precedence	Value	Bit Setting
Internetwork control	6	110
Network control	7	111

(Table Page 2 of 2)

**Quality of Service Bit Settings**

Following the three precedence bits, three bits define the quality of service assigned to each precedence. As shown in the following table, these bit settings instruct network routers about the packet's desired throughput, reliability, and delay. As with the precedence values, the quality of service values are inherently arbitrary. For example, how a network handles a packet marked as "low delay" depends on how the network measures congestion, as well as the rules that the network administrator has defined for handling "low delay" packets.

**Quality of Service Bit Settings**

Service Category	Bit Setting
Delay	0-normal delay 1-low delay
Throughput	0-normal throughput 1-high throughput
Reliability	0-normal reliability 1-high reliability

**Configuring Differentiated Services**

Follow the next procedure to configure differentiated services for streaming media IPv4 packets.

► **To define differentiated services:**

1. In Helix Administrator, click **Transport Settings>Differentiated Services**.
2. In the top set of fields, you can set the precedence and quality of service for streaming media packets in the RTP and the RDT formats. In the bottom set of fields, you can define the precedence and quality of service for packets of the RTSP control protocol. These settings do not affect packets for other protocols, such as MMS and HTTP.

**For More Information:** For more on RTP and RDT, see “Packet Formats” on page 122. For information on RTSP, see “Application-Layer Protocols” on page 121.

3. For a chosen protocol, select the packet precedence in the **Precedence** pull-down list. As described in “Precedence Bit Settings” on page 57, your network needs to define the characteristics for handling this precedence.
4. After setting a precedence for the packets, define their quality of service in the **Delay**, **Throughput**, and **Reliability** pull-down lists. As described in “Quality of Service Bit Settings” on page 58, your network needs to define the rules for handling the selected quality levels.
5. Click **Apply**.
6. Restart Helix Proxy.



## RATE CONTROL

This chapter describes how to configure Helix Proxy for server-side rate control. Using rate control, Helix Proxy can vary the streaming rate of a multi-rate clip to compensate for network congestion. This helps to minimize rebuffering by adjusting the streaming rate for the network's current conditions.

### Server-Side Rate Control Basics

Using server-side rate control, Helix Proxy can adjust the streaming bit rate of an on-demand clip or a pull-split broadcast streamed to a media player based on the player's buffer status. This allows Helix Proxy to compensate for fluctuating bandwidth that may occur due to network congestion.

### Rate Control Basic Requirements

Use of the server-side rate control feature depends on three general conditions:

- multi-rate clip or broadcast

Helix Proxy can shift streaming rates only if the requested clip or broadcast is encoded in a multi-rate format. The section “Media Formats Used with Server-Side Rate Control” on page 63 lists the acceptable formats.

- buffer information

Helix Proxy must have information about the media player's buffering characteristics. As described in “Buffer Modeling” on page 75, Helix Proxy adjusts streaming rates based on a model of the player's buffer state.

- status reports

Helix Proxy must receive periodic status reports from the media player to gauge network congestion and ensure that its buffering model is accurate. For information, refer to “Receiver and Sender Reports” on page 72.

## Types of Rate Control

When Helix Proxy streams a multi-rate clip or broadcast, it uses one of two types of server-side rate control, or the client-side SureStream rate control. The following sections explain these different streaming methods.

**Note:** If a media player does not meet the conditions required for any type of rate control, Helix Proxy can still stream a multi-rate clip or broadcast to the player. However, it does not shift the streaming rate during the presentation. Network congestion is therefore more likely to cause the media player to rebuffer the presentation.

### 3GPP Rate Adaptation

3GPP rate adaptation is a standards-based method of server-side rate control available to some 3GPP clients. It uses the standards-based RTP packet format. Helix Proxy can use 3GPP rate adaptation for the following clients:

- 3GPP Release 6 players that support 3GPP rate adaptation and NADU reports for RTP streaming. These players must send RTCP receiver reports as expected by  $b=RR$ . Helix Proxy can provide more efficient rate adaptation to players that also utilize the RTSP 3GPP-Link-Char header.
- 3GPP Release 5, PSS-compliant media players that implement the Annex G video buffer model. These players must send RTCP receiver reports as expected by  $b=RR$ .
- Mobile RealPlayers or Helix DNA-based players that support 3GPP Release 5 or Release 6 media types, as well as the RTP packet format.

### Helix Rate Adaptation

Helix rate adaptation is available for some RealNetworks media players, as well as players based on Helix DNA. Helix Proxy supports Helix Adaptation for the following media players:

- Desktop RealPlayer 11

**For More Information:** See “Enabling Rate Control for Desktop RealPlayers” on page 70.

- Mobile RealPlayers or Helix DNA-based players that implement Helix rate adaptation with RDT version 3 for RDT streaming, and provide RDT feedback reports.

**Tip:** Some of these players can switch to 3GPP rate adaptation if you turn off Helix rate adaptation support or disable the use of the RDT packet format.

**For More Information:** For information on RDT, refer to “Packet Formats” on page 122.

### SureStream Rate Adaptation

SureStream rate adaptation is the older, client-side method of rate adaptation used by desktop RealPlayers that predate RealPlayer 11. It requires the use of a multi-rate SureStream clip (RealAudio or RealVideo) and the RDT packet format. Helix Proxy switches to this type of rate control automatically for media players that support SureStream but not server-side rate control.

**Note:** Because the client makes rate shifting decisions when using SureStream rate control, none of the information contained in this chapter applies to this method of rate adaptation.

## Live Broadcasts and Rate Control

Rate control works for live, pull-split broadcasts streamed to any media player that supports Helix or 3GPP rate adaptation. Media players using 3GPP rate adaptation must also send the 3GPP-Link-Char header to support upshifting. Players that do not support either of these adaptation methods receive a single-rate stream for the duration of the broadcast.

## Media Formats Used with Server-Side Rate Control

For rate control to function, Helix Proxy must deliver a clip that includes multiple streams encoded at different bit rates. For RealNetworks desktop media players, you can use SureStream.

## SureStream Clips for RealNetworks Players

A SureStream clip encodes multiple streams at different bit rates. The mechanism used to control the shifting of bit rates during playback varies according to the media player version:

- For desktop RealNetworks media players earlier than RealPlayer 11, Helix Proxy uses SureStream rate adaptation. This causes RealPlayer to request the upshifts and downshifts.
- For RealPlayer 11 and later, Helix Proxy manages stream shifting based on the Helix rate adaptation model. This model allows delivery of all existing SureStream clips.

## Media Encoding Guidelines

When you encode a multi-rate clip or broadcast for server-side rate control, Helix Proxy allows upshifting to the next higher stream rate only if that rate is no more than a certain percentage above the speed of the current rate. This is called the *maximum oversend rate*.

Suppose that the maximum oversend rate is 200 percent. For a clip encoded with 32, 64, 128, and 256 Kbps streams, the media player can upshift through all streams because no rate is more than 200 percent of the preceding rate. If the clip contains just 64 and 256 Kbps streams, however, the media player cannot upshift from the 64 Kbps stream.

**For More Information:** The section “Using the Default Profile or Preset Values” on page 94 lists the max oversend rate used when you choose a default streaming profile using Helix Administrator. You can also set a value manually in a user agent profile with the `MaxOversendRate` variable. For more information, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

## User Agent Profiles

You configure rate control by defining a series of profiles. Each profile is closely tied to the streaming capabilities of a media player or group of players. For instance, you might define one profile for desktop RealPlayers and one for Helix-based clients. Each profile reflects the player’s streaming environment, such as its typical network speed and its media buffer size.

## UAS Profiles

Helix Proxy can write each profile to a separate user agent settings (UAS) file. This is an XML-formatted text file that uses the extension `.uas`. Helix Proxy stores the `.uas` files in the `ClientProfiles` subdirectory of its main installation directory. When you use Helix Administrator to create a profile, the file name follows the profile name. For example, if you create a profile called Helix DNA Player, the UAS file is named `Helix DNA Player.uas`. The UAS file can use any unique file name, however.

## Configuration File Profiles

Creating separate UAS files is recommended because of its convenience. However, you can also store profile information in the Helix Proxy configuration file. Each profile is created within the configuration file's `UserAgentSettings` list.

**For More Information:** For details about the profile syntax, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

## RDF Files

A media player may have an associated Resource Description Framework (RDF) file. When the media player contacts Helix Proxy, it indicates the Internet location of its RDF file, which typically resides on a Web server hosted by the media player's manufacturer.

Like the UAS files generated by Helix Proxy, the RDF file describes the player's streaming capabilities. Helix Proxy provides the option to use the RDF file or ignore it. If you choose to use it, Helix Proxy downloads the RDF file and integrates some of the RDF values into the client's UAS file.

**Note:** A media player can send Helix Proxy the RDF file location in the `x-wap-profile` header of an HTTP request or any RTSP message.

**Tip:** To ensure that Helix Proxy can retrieve RDF files, configure your firewall to allow Helix Proxy to contact external HTTP servers on TCP port 80. For more about firewalls, see Chapter 9.

## Profile Value Inheritance

A rate control profile (a *child* profile) can inherit values from another profile (a *parent* or *base* profile). For example, a profile might inherit values from the Default profile stored in the Helix Proxy configuration file. To these values, it adds its own values for variable not defined within the Default profile. A profile stored in the configuration file can inherit values from a profile defined within a UAS file, and vice versa.

### Inheritance Chains

Inheritance can extend across any number of profiles. For example, profile A can inherit values from profile B, which inherits values from profile C, and so on. A single profile can have only one base profile, however. If a profile and its base profile set conflicting values for the same variable, Helix Proxy uses the value from the child profile.

### Profile Configuration Errors

In most situations, Helix Proxy compensates for profile configuration problems:

- missing configuration values

It is OK for a configuration variable **not** to be defined in a child profile or any profile from which the child inherits values. In this case, Helix Proxy applies the default value for the undefined variable.

**For More Information:** For information about rate control default values, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

- missing base profile

If a profile inherits values from a base profile that is missing or misconfigured, these values revert to their defaults. Helix Proxy writes an error to its error log warning about a missing base profile.

**For More Information:** See “Basic Error Log” on page 156.

- looping inheritance

An example of an inheritance loop is when profile A inherits values from profile B, which inherits values from profile A. In this case, if profile A is

used, Helix Proxy treats profile B as missing and writes an error message about an inheritance loop.

## User Agent Strings

Helix Proxy uses a media player's user-agent string to determine which rate control profile to use for the player. Formats for user-agent strings vary widely, and you should know the user-agent strings sent by individual media players when defining profiles. Typically, the user-agent string identifies the application, version, and compatibility:

*application/version (compatibility)*

For example, the following user-agent string is for a Helix-based media player: HelixDNAclient/10.0.0.0 (symbian)

The user-agent string can also identify the player's capabilities, such as which stream formats it can render. When the media player connects to Helix Proxy, it passes its user-agent string in the RTSP connection headers. For each rate control profile, you define one or more user-agent strings that the profile matches. Matches can be partial or exact, depending on your needs.

### Exact Profile Matching

You can set a profile to match a media player's user-agent string exactly by adding an equals sign (=) to the beginning of the string:

=HelixDNAclient/10.0.0.0 (symbian)

In this case, the user-agent string listed in the profile must match the user-agent string sent by the client character-for-character in its entirety. The match is also case-sensitive.

### Partial String Matches

A profile might include a partial user-agent string. Such a string, for instance, might match all media players of a certain brand. It does this by identifying a common element of the user-agent string sent by all models of this brand. For example, the following user-agent string matches all version 10 Helix DNA clients:

HelixDNAclient/10

The preceding string matches the user-agent strings sent by all Helix DNA version 10 clients (including versions 10.1, 10.2, and so on) running on any

operating system. Unlike exact user-agent string matching, partial matching is **not** case-sensitive.

### Multiple Profile Sorting

Because of partial matching, more than one rate control profile may match a user-agent string sent by a media player. In this case, Helix Proxy selects the profile to use by applying the following criteria in order:

1. Exact match.

If a profile provides an exact match to the user-agent string, Helix Proxy uses that profile and ignores profiles that provide partial matches.

2. Most complete match.

If multiple profiles are partial matches, Helix Proxy chooses the profile that provides the greatest number of exactly matched characters.

3. Last child match.

If profiles inherit values, the child profile furthest from the parent is used. For example, suppose that profile B inherits values from profile A, and profile C inherits values from profile B. If profiles B and C equally match a user-agent string, Helix Proxy chooses profile C.

4. First found.

If all of the preceding criteria still result in multiple, matching profiles, Helix Proxy uses the first of these profiles it found as it searched for the matches.

## Profiles Shipped with Helix Proxy

Helix Proxy includes a number of predefined profiles, as well as a default profile.

### Default Profile

The Helix Proxy configuration file defines a profile named Default. The server uses this profile if the user-agent string sent by the client does not match any other profiles defined within the configuration file or the separate UAS files. You can use the Default profile as the base profile when defining additional streaming profiles.

**For More Information:** The section “Using the Default Profile or Preset Values” on page 94 describes the values set by the Default profile.

### Predefined User-Agent Files

The following are the user-agent profiles included as separate .uas files in the ClientProfiles subdirectory of the Helix Proxy installation directory. These profiles inherit values from the Default profile, then set additional values specific to the client type:

- DesktopPlayers.uas – QuickTime and VLC Desktop Clients
  - QuickTime
  - VLC media player
- Motorola.uas – Motorola Mobile Clients
  - MOT-A820/00.00.00 MIB/2.1
  - MOT-MOTORAZRV9
- NokiaR1M.uas – RealNetworks Mobile Clients
  - RealOnePlayer
  - RealMedia Player/3
  - RealMedia Player/s
  - RealMedia Player (epoc\_
  - RealMedia Player HelixDNAClient/10.0.0.6839 (S60 HX cvs\_cays221\_2008
  - RealMedia Player HelixDNAClient/10.0.0.6839 (S60 HX cvs\_cays221\_2009
- PVPlayer.uas – PacketVideo Clients
  - PVPlayer 3.0 PVPlayer/3.1
  - PVPlayer/3.4
  - PVPlayer 3.4
- QtvPlayer.uas – LG, Samsung, and Qualcomm QtvPlayer
  - QtvPlayer
  - RealMedia Player (HelixDNAClient)/10.0.0.0 (Qualcomm Inc.)
  - QtvPlayer (org=SS; model=13; terversion=11
  - SAMSUNG-SGH-A707

- RealPlayer.uas – RealNetworks Desktop Clients
  - RealMedia Player (HelixDNAClient)
  - RealMedia Player HelixDNAClient
  - RealMedia Player (HelixDNAClient)/10.0.0.7217 (win32)
  - RealMedia Player HelixDNAClient/10.0.1.338 (win32)
  - RealMedia Player HelixDNAClient/10.0.1.564 (win32)

### Enabling Rate Control for Desktop RealPlayers

Desktop RealPlayers use a version of client-side rate control. However, RealPlayer 10.5 and later support server-side rate control if you enable it. To do this, open the RealPlayer.uas file in the ClientProfiles subdirectory. In the profiles for RealMedia Player versions 10.0.0.7217 and higher, change the values of UseServerSideRateAdaptation and UseServerSideRateAdaptationForLive from never to clientselect:

```
<Var UseServerSideRateAdaptation="clientselect" />  
<Var UseServerSideRateAdaptationForLive="clientselect" />
```

**Note:** RealNetworks recommends that you do this only in consultation with RealNetworks support staff.

## Streaming Rate Selection

The following sections explain the user agent settings that affect how much bandwidth Helix Proxy may use for a client connection, and how Helix Proxy selects the initial streaming rate for a multi-rate clip.

### Maximum Sending Rate

Using Helix Administrator, you can set a maximum bandwidth for a profile. This is optional, and the default maximum is 10 million bits per second. It is useful to lower the maximum, however, for networks that limit the bandwidth of each client connection. In this case, Helix Proxy considers any stream that exceeds the maximum sending rate as unstreamable. It excludes these streams when choosing the initial streaming rate or upshifting to higher rates.

**For More Information:** In the user agent settings, the MaxSendRate variable controls this sending rate. For details,

refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

## Initial Rate Selection

When a media player requests a multi-rate clip, Helix Proxy evaluates several user agent values to determine which encoding to stream. Once it has established the initial rate criterion to use, Helix Proxy chooses the encoding that is closest to the specified bandwidth without exceeding it. The following are the user agent values, listed in order from highest to lowest priority, that Helix Proxy evaluates to select the initial streaming rate:

1. Helix Proxy first evaluates query string parameters appended to the request URL. The optional `ir=rate` parameter indicates the preferred bit rate in bits per second. The following example shows the query parameter in an RTSP request:

```
rtsp://helixserver.example.com/video1.rm?ir=28000
```

The next example shows the `ir` parameter in a Ramgen request:

```
http://helixserver.example.com/ramgen/video1.rm?ir=28000
```

2. If no `ir` parameter is present in the request URL, Helix Proxy next determines the initial streaming rate based on the value of the `Subscribe` header sent as part of the RTSP `SET_PARAMETER` request. This value is sent only by RealNetworks and Helix-based media players.
3. If `UseRel6InitialRateSelection` is set to 1 (true) in the media player's user agent settings, the next evaluated criterion is the stream choice presented in the `stream.id` of the RTSP `SETUP` request.

**For More Information:** This variable must be set directly in the user agent settings. For more on the `UseRel6InitialRateSelection` variable, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

4. For 3GPP clients, Helix Proxy uses the guaranteed bandwidth value (GBW) returned by the client in the `3GPP-Link-Char` header. Not all 3GPP clients provide this header, however.
5. Helix Proxy evaluates the `Bandwidth` header sent by the client in an RTSP `OPTIONS`, `DESCRIBE`, `SETUP`, `SET_PARAMETER`, or `PLAY` request. This header is optional, and is used to indicate the estimated bandwidth available to the client in bits per second.

6. If the preceding options do not produce a viable initial rate, Helix Proxy uses the default media rate specified in the client's user-agent settings. The variable `DefaultMediaRate` in the rate adaptation section provides this value.

**For More Information:** You cannot set `DefaultMediaRate` using Helix Administrator. You must set it in the user agent profile manually. Refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

7. If `UseRel6InitialRateSelection` is set to 0 (false) in the media player's user agent settings, the last evaluated criterion is the `stream.id` in the RTSP SETUP request.

## Initial Oversend Rate

Once Helix Proxy selects the initial streaming rate, it may begin to stream the clip faster than the initial rate. The purpose is to fill the media player's buffer quickly so that the player can start to render the stream as soon as possible. By default, Helix Proxy may transmit the initial stream at 100 to 400 percent of the actual streaming speed, depending on factors such as the media player capabilities, the stream bandwidth, the maximum sending rate, and so on.

**For More Information:** The section "Using the Default Profile or Preset Values" on page 94 explains the maximum, initial oversend rate selected when you choose a default streaming profile using Helix Administrator. You can also set the `InitialOversendRate` variable directly in the user agent settings. For more on this, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

## Receiver and Sender Reports

For server-side rate control to function accurately, Helix Proxy must receive periodic updates from the media player. These updates, or *receiver reports*, allow Helix Proxy to synchronize its buffer model with the actual buffer state experienced by the client. Using this information, Helix Proxy changes the streaming rate as necessary to compensate for network congestion.

**Note:** The Helix rate adaptation method uses a proprietary feedback mechanism implemented in the RDT packet format.

Information in the following sections pertains only to the 3GPP rate adaptation method, which is based on the RTP packet format.

**For More Information:** For more on receiver reports, refer to RFC 1889 at <http://www.ietf.org/rfc/rfc1889.txt>.

## Report Frequency

For rate control to function, a media player must send RTCP receiver reports to Helix Proxy. The report frequency is typically less than one second, but may be up to five seconds. The shorter the reporting interval, the greater Helix Proxy's ability to adjust streaming rates in a timely manner.

### Receiver Report Rate

To send receiver reports at an appropriate interval, the media player should recognize the `b=RR` directive set in the SDP file. This directive determines the frequency of receiver reports. The default action is to request the client to reserve the amount of bandwidth equal to two percent (2%) of the initial streaming rate. Helix Proxy can also request a set amount of bandwidth regardless of the initial streaming rate.

Suppose that the initial streaming rate is 32 Kbps and the receiver report rate is two percent. Helix Proxy requests the media player to use 0.64 Kbps ( $32 \times 0.02$ ) of the connection bandwidth for receiver reports. This gives the player the ability to transmit 80 bytes of feedback per second. If a receiver report is 160 bytes in size, the receiver report frequency is one feedback report every two seconds.

When receiver report rates are defined as percentages of initial bandwidth, the bandwidth reserved for the reports increases along with the initial streaming rate. This means that higher-bandwidth clips report client feedback more frequently. This action is appropriate because, at higher bandwidths, the media player consumes data from its buffer faster, requiring that Helix Proxy compensate for network congestion more quickly.

**Tip:** After the bandwidth reserved for receiver reports has been set, it does not change if the streaming rate shifts up or down during the session.

**For More Information:** As explained in the section "Using the Default Profile or Preset Values" on page 94, Helix

Administrator allows you to select a streaming profile that sets a receiver report bandwidth. You can also set a value manually by changing the `RTCPRRrate` variable in the user agent settings. For details, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

### Sender Report Rate

Helix Proxy also sends the media player sender reports. The report rate is set by the `b=RS` directive set in the SDP file. The default is to reserve for sender reports the amount of bandwidth equal to one percent (1%) of the initial streaming rate. Helix Proxy can also request a set amount of bandwidth regardless of the initial streaming rate. As with receiver reports, the reserved bandwidth does not change during the session.

**For More Information:** The section “Using the Default Profile or Preset Values” on page 94 lists the sender report bandwidths that you can select using Helix Administrator. You can also set a value manually by changing the `RTCPRSrate` variable in the user agent settings. For details, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

### NADU Feedback

Helix Proxy supports the inclusion of NADU (Next Application Data Unit) information in receiver reports. The NADU signaling method provides feedback on the client buffer state. Helix Proxy uses this information to keep its server-side model of the player’s buffer accurate. By default, Helix Proxy requests media players that support NADU to include NADU information in each receiver report.

**For More Information:** You can decrease the NADU reporting frequency by changing the `NADUReportFreq` value in the Helix Proxy configuration file. This variable sets a global value that affects all media players. For details, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

### Receiver Report Timeout

If ten seconds elapse and a new receiver report has not arrived, Helix Proxy disables rate control for the media player. It immediately shifts to the lowest streaming rate, and does **not** resume rate control even if a new receiver report

arrives. The downshift to the lowest rate occurs even if you have configured downshifting to occur in stages, as described in the section “Stepwise Downshifting” on page 81.

You cannot change the timeout behavior using Helix Administrator. However, you can modify it using variables in the media player’s user agent settings:

- The `FeedbackTimeout` variable sets the feedback timeout period in milliseconds.
- You can disable downshifting to the lowest media rate by setting the value of the `DownshiftOnFeedbackTimeout` variable to 0. In this case, Helix Proxy continues to stream at the current rate when a receiver report timeout occurs.

**For More Information:** Refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

## Buffer Modeling

To implement server-side rate control, Helix Proxy maintains a model of each player’s data buffer. This allows it to track throughout the streaming session the amount of data the player has available to render. Using receiver reports, it constantly updates its model to synchronize it with the player’s actual state. Helix Proxy can then upshift or downshift the streaming rates as needed. The following sections explain the basics of how Helix Proxy models a media player’s buffer.

### Data Size Versus Playback Time

The amount of data in a media player’s buffer is called its *buffer depth*. There are two ways to measure buffer depth: bits of data or milliseconds of playback time. Correspondingly, Helix Proxy allows you to set parameters that affect buffer modeling in terms of the amount of data in the buffer, or the length of the clip playback time that’s buffered.

Because a media player’s data buffer has a fixed size, it is natural to think of buffering values in bits and bytes. However, this may not be advantageous for certain settings, such as a downshift depth (see “Downshifting” on page 79). Suppose that you set the buffer depth at which to downshift as 1/16 of a buffer capacity of 256 Kilobytes (KB). This means that a downshift occurs when the buffer drops to 16 KB of data.

For a 128 Kilobit-per-second (Kbps) media stream, 16 KB represents one second of buffered playback time. In this case, it is a good idea to downshift to a slower streaming speed. For a 32 Kbps stream, however, 16 KB is four seconds of buffered playback time. At a relatively slow streaming rate of 32 Kbps with four seconds of buffered playback, downshifting is not necessary. Alternatively, you could set the downshift point at 2000 milliseconds. This means that the presentation downshifts when the buffer contains no more than two seconds worth of media. For a 128 Kbps stream, this is 32 KB of data. For a 32 Kbps stream, this is 8 KB of data.

Using millisecond values for certain buffer depths such as the upshift point may cause the inverse problem. Suppose that you set the upshift point at 8 seconds. For a 256 Kbps stream, the upshift occurs only when the 256 KB buffer fills to its maximum capacity. Although the upshift will still occur, the quality of playback will be degraded. Setting an upshift point as a percentage value makes more sense in this case.

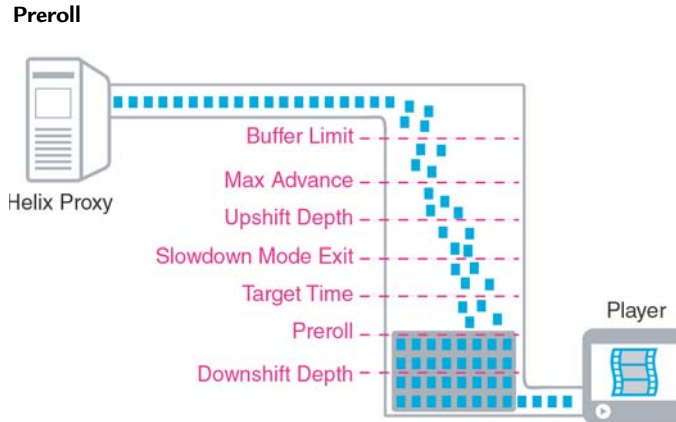
## Buffer Size

Each media player has a maximum buffer size typically defined in its RDF file. Or, you can use Helix Administrator to set this size directly in the UAS file as the value for **Video Predecode Buffer Size**. By default, Helix Proxy sets a limit on buffer usage as 98 percent of the maximum capacity. It automatically limits some other buffer variables to this usage limit. For example, if the max advance level exceeds the buffer usage limit, Helix Proxy automatically resets the max advance level to be equal to the usage limit.

**For More Information:** You can change the usage limit value by setting the `BufferUsageLimit` variable in the user agent settings. For details, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

## Preroll

Before a media player begins to render a stream, it buffers a certain amount of the stream data. This initial buffering, called *preroll*, may vary from clip to clip. The following figure illustrates a media player's buffer filling with data until the necessary preroll depth is reached. At this point, the player begins to render the stream.



When modeling the player's buffer, Helix Proxy assumes that playback begins once the preroll has been met. However, as the following sections explain, certain media players may start playback at a point other than the clip's standard preroll value.

### Encoded Preroll Value

The most commonly observed preroll is a value encoded directly into the clip. The codec used to create the clip sets this preroll value, which is usually measured in milliseconds. Many media players respect the preroll value, buffering data until the preroll level is reached. Because preroll can vary from clip to clip, the initial buffering period may change for a media player depending on the content it plays.

### Minimum Preroll In SDP

A media player's user agent settings can include a `MinPreroll` variable that specifies a preroll value in milliseconds. If this variable is present, Helix Proxy advertises this minimum preroll value in the SDP file delivered to the player. If the player respects the SDP preroll value, it uses the larger of the two preroll values. That is, if the minimum preroll value advertised in the SDP file is larger than the encoded preroll value, the SDP value is used, and vice versa.

**Note:** Not all media players observe the preroll value in the SDP. RealNetworks and Helix-based players respect the SDP-derived preroll. Many 3GPP players do not, however. Be sure that you know the player characteristics before setting this variable.

**For More Information:** The section “Using the Default Profile or Preset Values” on page 94 explains the minimum preroll values set when you choose a default streaming profile using Helix Administrator. You can also set the MinPreroll value manually in the user agent settings. For details, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

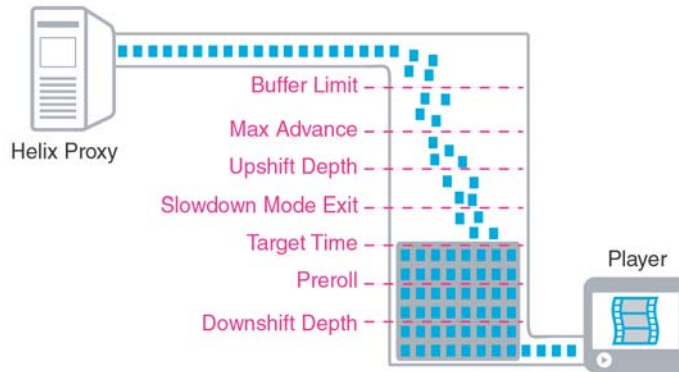
### Client-Specified Buffer Period

Some media players buffer a specific number of milliseconds for each stream as the preroll, regardless of the preroll value encoded in the stream or advertised in the SDP. In this case, you can set the ClientBufferPeriod variable in the user agent settings to match the player’s behavior. Helix Proxy then uses this value in its buffer model to estimate when the player begins media playback.

**For More Information:** You must set the ClientBufferPeriod value manually in the user agent settings. For instructions, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

### Target Time

Media players that support Helix rate adaptation or 3GPP rate adaptation provide Helix Proxy with a Target-Time value. This is the number of milliseconds of stream playback time that the media player prefers to have buffered until playback ends. The target time can differ for each media player. It may be above or even below a clip’s encoded preroll time. The following figure illustrates Helix Proxy maintaining a player’s buffer at a target time level that is higher than the clip’s preroll requirement.

**Target Time**

No user agent variables directly affect the target time, which is determined entirely by each media player. Various user agent variables, however, can set the stream upshift and downshift points relative to the reported target time. If a media player does not report a target time, Helix Proxy attempts to maintain the buffer at the preroll level or higher. It then bases the upshift and downshift points on the preroll value.

**Tip:** Because the client-specific Target-Time value reflects the media player's buffering capability better than a stream-specific preroll value, stream shifting based on Target-Time is likely to be more accurate than that based on media preroll.

## Downshifting

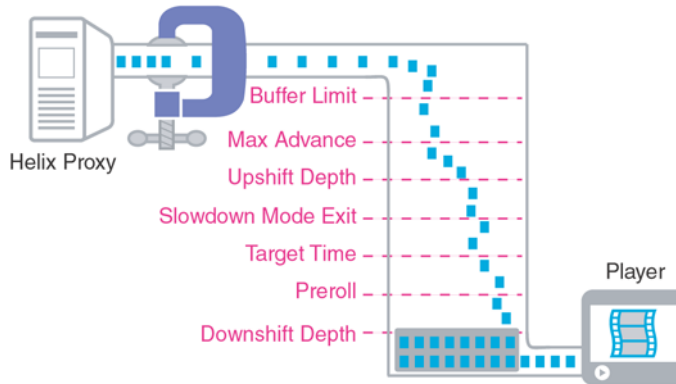
When a clip is encoded at multiple bandwidths, Helix Proxy can *downshift* to a stream encoded at a lower bandwidth (less data per second). This occurs when the media player's buffer begins to drain. A low buffer indicates that the player is consuming packets faster than they are being delivered by the network.

Downshifting causes the player to consume data from its buffer at a slower pace, thereby allowing the buffer to refill by bringing the rate of packet consumption in line with the rate of packet delivery. This, in turn, helps prevent rebuffering, an undesirable condition in which the player must halt the presentation to refill an empty buffer.

If the network conditions that necessitated the downshift clear, the player's buffer may fill to a level that allows Helix Proxy to upshift back to the original streaming rate, or even to a higher streaming rate. The following figure illustrates network congestion that causes packets to be delivered more slowly

than they are rendered. Once the player buffer reaches its downshift depth, Helix Proxy switches the stream to a slower encoding.

### Downshift Depth



### Downshift Depth Settings

Helix Administrator provides two separate fields in which you can specify the downshift depth. The two values affect different types of media players.

**Tip:** It is OK to define values for both **Downshift Depth** and **Target-Time Downshift Depth** in the same profile. Helix Proxy automatically chooses which value to use based on the media player's capabilities.

### Downshift Depth

The **Downshift Depth** value works with RealPlayer 10 and earlier, and older Helix-based media players. It defines a downshift depth value based on the media preroll, which is encoded in the stream and may vary for each stream. The default depth is 80 percent of the preroll value. You can also set the downshift depth as a certain number of milliseconds of buffered media playback, such as 1000 milliseconds. In this case, the clip's preroll value has no effect on downshifting.

**Note:** You can set this downshift depth using Helix Administrator. Or, you can change the value of the DownshiftDepth variable manually in the player's user agent settings.

## Target-Time Downshift Depth

The **Target-Time Downshift Depth** value applies to RealPlayer 11 and later, and newer Helix-based media players. It sets the downshift depth as a percentage of the Target-Time value reported by the media player. The default value is 80 percent of Target-Time. You can also set this downshift depth to a certain number of milliseconds of media buffered, such as 1000 milliseconds. In this case, the reported Target-Time value has no effect on downshifting.

**Note:** You can set this downshift depth using Helix Administrator. Or, you can change the value of the TargetTimeDownshiftDepth variable manually in the player's user agent settings.

## Stepwise Downshifting

Stepwise downshifting allows Helix Proxy to step down through each encoded rate in the clip. This feature is **disabled** by default, which causes Helix Proxy to downshift immediately to the most appropriate bit rate, bypassing any intermediate streaming speeds. The default behavior sacrifices the higher quality of any intermediate bandwidths in an attempt to ensure that the media player's buffer does not drain.

You may want to enable stepwise downshifting if your network provides relatively stable bandwidth for each client connection. In this case, Helix Proxy may go through a longer cycle of downshifting to reach the highest sustainable speed. This keeps the quality of the rendered media as high as possible on each downshift. However, this also increases the chance that the media player may need to rebuffer the presentation if the available bandwidth drops significantly.

**For More Information:** To enable stepwise downshifting, change the value of the EnableStepwiseDownshift variable in the media player's user agent settings to 1. For more information, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

**Note:** Helix Proxy waits a certain time between successive rate shifts. The section "Using the Default Profile or Preset Values" on page 94 explains the stream switch delay selected when you choose a default streaming profile. You can also modify the

value of the StreamSwitchDelay variable in the player's user agent settings.

## Client-Initiated Downshift

If a media player supports the RTSP 3GPP-Link-Char or Bandwidth header, it can send information about its network link characteristics to Helix Proxy over the RTSP channel. By default, Helix Proxy allows the media player to initiate a downshift in the streaming rate by using one of these protocol headers. This allows downshifting to occur immediately, without Helix Proxy having to wait for the next client receiver report.

**For More Information:** You can turn off support for this feature by setting the EnableClientReportedDownshift variable in the player's user agent settings to 0. For more information, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

## Streaming Below the Lowest Media Rate

By default, the lowest rate at which Helix Proxy transmits data to the media player is the lowest streaming rate encoded in the clip. However, you can set Helix Proxy to deliver the slowest stream at a rate even slower than its encoded speed. This allows Helix Proxy to transmit data to the player during periods of extremely high network congestion. Although the player will most likely rebuffer the stream once the congestion clears, it may have received enough data to shorten the rebuffering period.

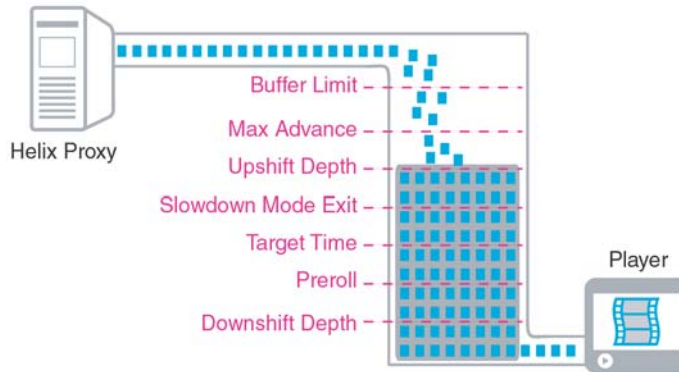
**For More Information:** To enable streaming below the lowest encoded rate, set the value of the DeliverBelowLowestMediaRate variable in the media player's user agent settings to 1. For details, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

## Upshifting

*Upshifting* means that Helix Proxy switches to a media stream that is encoded at a higher bandwidth (more data per second). This occurs when the media player's buffer begins to fill. A nearly full buffer indicates that the player is consuming packets from its buffer slower than they are being delivered.

Upshifting causes the player to consume data from its buffer at a faster pace. This helps to synchronize packet consumption from the buffer with packet delivery by the network. By upshifting to a higher data rate, Helix Proxy delivers a higher-quality user experience, and prevents data packets from being lost because the media player's buffer was too full to accept them.

### Upshift Depth



## Upshift Depth Settings

Helix Administrator provides two separate fields in which you can specify the upshift depth. The two values affect different types of media players.

**Tip:** It is OK to define values for both **Upshift Depth** and **Target-Time Upshift Depth** in the same profile. Helix Proxy automatically chooses which values to use based on the media player capabilities.

### Upshift Depth

The **Upshift Depth** value works with RealPlayer 10 and earlier, and older Helix-based media players. It defines an upshift depth based on the media preroll, which is encoded in the stream and may vary for each stream. The default depth is 400 percent of the preroll value. You can also set the upshift depth as a certain number of milliseconds of buffered media playback, such as 4000 milliseconds. In this case, the clip preroll has no effect on upshifting.

**Note:** You can set this upshift depth using Helix Administrator. Or, you can change the value of the

UpshiftDepth variable manually in the player's user agent settings.

### Target-Time Upshift Depth

The **Target-Time Upshift Depth** value applies to RealPlayer 11 and later, and newer Helix-based media players. It sets the upshift depth as a percentage of the Target-Time value reported by the media player. The default value is 400 percent of Target-Time. You can also set the upshift depth as a certain number of milliseconds of buffered media playback, such as 4000 milliseconds. In this case, the reported Target-Time value has no effect on upshifting.

**Note:** You can set this upshift depth using Helix Administrator. Or, you can change the value of the TargetTimeUpshiftDepth variable manually in the player's user agent settings.

### Stepwise Upshifting

Stepwise upshifting causes Helix Proxy to step up in sequence through each encoded rate in the clip until it finds the maximum sustainable rate. This feature is enabled by default, causing Helix Proxy to upshift more gradually through the intermediate streaming speeds. The default behavior produces conservative upshifting that attempts to ensure that each upshift is sustainable.

**For More Information:** You can disable stepwise upshifting by changing the value of the EnableStepwiseUpshift variable in the player's user agent settings to 0. For details, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

**Note:** Helix Proxy waits a certain time between successive rate shifts. The section "Using the Default Profile or Preset Values" on page 94 explains the stream switch delay selected when you choose a default streaming profile. You can also set the value of the StreamSwitchDelay variable in the player's user agent settings.

## Client-Initiated Upshift

Media players that support the RTSP 3GPP-Link-Char or Bandwidth header can send information about network link characteristics to Helix Proxy over the RTSP channel. By default, Helix Proxy allows the media player to initiate an upshift in the streaming rate by using one of these protocol headers. This allows upshifting to occur immediately, without Helix Proxy having to wait for two consecutive receiver reports.

**For More Information:** You can turn off support for client-initiated upshifting by setting the `EnableClientReportedUpshift` variable in the player's user agent settings to 0. For details, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

## Slowdown Mode

Helix Proxy can use a feature called *slowdown mode* to help ensure that a media player's buffer does not overflow on an upshift. The user agent settings that control slowdown mode typically do not need to be changed from their default values. However, you need to enable slowdown mode, and may want to change the max advance level if a media player frequently experiences packet loss during upshifting.

**For More Information:** You enable slowdown mode by setting the value of the `RateManagerType` variable in the player's user agent settings to `unblocking`. For details, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

## Slowdown Mode During Upshifting

During upshifting, Helix Proxy attempts not to overload a media player's buffer. For example, suppose that a media player is receiving a 32 Kbps stream. Network conditions are good and the player's buffer has filled to its upshift depth, which equals six seconds of playback time. Once the upshift depth is reached, Helix Proxy switches to a 64 Kbps encoding of the stream. This upshift may overload the player's buffer, however.

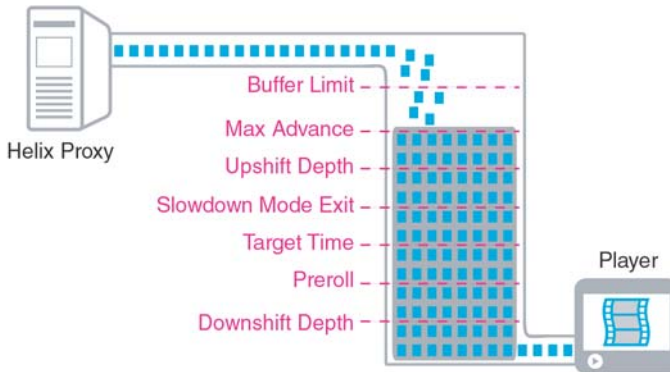
Suppose that the buffer limit is 10 seconds of data at 32 Kbps. The buffer already contains six seconds of the 32 Kbps stream yet to play. This leaves four

seconds at 32 Kbps, or only *two seconds* of available buffer space at 64 Kbps. At the faster data rate, the buffer may overflow with new 64 Kbps data before the player has been able to drain the existing 32 Kbps data. If this happens, some 64 Kbps packets may be lost to the network, lessening the quality of the playback when the player starts to render the 64 Kbps stream.

## Max Advance

The *max advance* value defines a buffer level that is usually above the upshift depth but lower than the buffer limit. By default, it is the buffer usage limit. If Helix Proxy determines that the buffer has reached the max advance level, it slows the amount of data it transmits to allow the buffer to drain. The following figure illustrates a player reaching its max advance state and entering slowdown mode.

### Max Advance Level



**Warning!** The max advance level should always be higher than clip preroll or the player's Target-Time value. The max advance level must also be higher than the upshift depth for upshifting to occur.

**For More Information:** You can change the max advance point by setting the MaxAdvance variable in the user agent settings. For details, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

## Limiting Stream Buffering with Max Advance

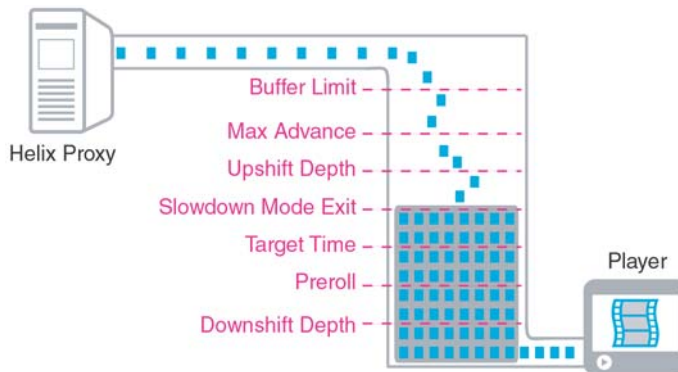
Helix Proxy always enters slowdown mode when a media player's buffer reaches its max advance level. This occurs even if upshifting is not possible because the highest-rate stream is already playing, or the clip is encoded at a single rate. Because you set a max advance value in milliseconds, you can use max advance to cause a media player to buffer only a certain amount of media playback time. If you set a player's max advance level to 3000 milliseconds, for example, you cause the media player to buffer approximately three seconds of its current stream.

## Slowdown Mode Exit

When it enters slowdown mode, Helix Proxy drops the data transmission rate to approximately 70 percent of the initial stream's natural streaming rate. This increases empty buffer space by allowing the buffer to drain to a level defined as the *slowdown mode exit*. Once the exit point is reached, data transmission resumes at the upshifted streaming rate.

By default, the exit point is the Target-Time value or the content preroll, whichever the media player uses. You can change the slowdown mode exit point to any value, however. Typically, the exit point falls between the values for the upshift depth and the downshift depth. The following figure illustrates reduced data transmission until the buffer reaches the slowdown mode exit point.

### Slowdown Mode Exit



**For More Information:** You can change the exit point by setting the `SlowdownModeExit` variable in the media player's user agent

settings. For details, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

**Tip:** Once it leaves slowdown mode, Helix Proxy waits for two consecutive receiver reports that indicate an upshift is possible. You can enforce a more conservative upshift behavior by increasing the value of the `RequiredUpshiftConfirmations` variable in the media player's user agent settings. For details, refer to the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

## Congestion Control

Helix Proxy can employ different methods of congestion control. These methods allow it to deliver a steady amount of data to a media player by compensating for various fluctuations in available bandwidth. Rate shifting between different streams in a clip occurs only on a persistent change in network bandwidth that exceeds the abilities of the congestion control mechanism. The default congestion control mechanism is TFRC.

**For More Information:** You can change the congestion control mechanism using the `UDPCongestionControlType` variable in the media player's user agent settings. The section "Enabling BCC" on page 96 explains how to set BCC as a profile's congestion control mechanism.

## TFRC

TCP Friendly Rate Control (TFRC) is the default means of congestion control when the UDP transport is used. It poses a problem, however, for some networks that contain large network buffers (around one Megabyte or larger). On such networks, TFRC may cause large round trip times, negatively affecting throughput.

**For More Information:** For information about TFRC variables, refer to the TFRC section in the rate control chapter of *Helix Proxy Configuration and Registry Reference*. For more about TFRC, refer to IETF RFC 3448 at <http://www.ietf.org/rfc/rfc3448.txt>.

## BCC

Binomial Congestion Control (BCC) can be used on networks with large buffers along the link path. It insures that correct congestion control response is maintained while not causing the round-trip time to increase to excessive levels. You should not choose BCC without first performing diagnostics on the network to determine if this type of congestion control is appropriate. BCC works only with the UDP transport.

**For More Information:** For information about BCC variables, refer to the BCC section in the rate control chapter of *Helix Proxy Configuration and Registry Reference*. You can learn more about BCC at <http://nms.lcs.mit.edu/papers/binomial-infocom01.html>.

## TCP

If a media stream is delivered using TCP rather than UDP, Helix Proxy relies on the native TCP congestion control mechanism, which is built into the TCP/IP network layer. The TCP transport is available to any media player that prefers it.

**Note:** If the user agent variable `PreferTCP` is set to 1, Helix Proxy uses TCP for media players that advertise support for the TCP transport.

**For More Information:** For information about TCP variables, refer to the TCP congestion control section in the rate control chapter of *Helix Proxy Configuration and Registry Reference*.

## Defining Rate Control Profiles

Using Helix Administrator, you can configure rate control profiles for various media players. You can create the profile in the Helix Proxy configuration file, or store it as a separate user agent file. For each profile, you can set the major rate control variables, such as those that control the upshift and downshift depths.

**Note:** You cannot set every rate control variable using Helix Administrator. As noted in the preceding sections, some variables must be defined manually in the user agent settings.

► To define a rate control profile:

1. In Helix Administrator, click **Proxy Setup>Media Delivery**.
2. Two pull-down menus allow you to enable or disable advertisement for the two methods of server-side rate control, which the section “Types of Rate Control” on page 62 explains. By advertising a rate control method, Helix Proxy indicates that it supports the method’s feedback mechanism. A media player may then choose to send feedback for one of the rate control methods based on its capabilities.
  - a. For **Advertise Helix Adaptation**, select Yes to enable or No to disable advertisement of the Helix rate adaptation method to all media players. Enabling this feature allows RealNetworks and Helix-based media players to send Helix rate adaptation feedback.

**Tip:** If you disable this feature, some RealNetworks and Helix-based media players may still be able to use the 3GPP rate adaptation.
  - b. For **Advertise 3GPP Adaptation**, select Yes to enable or No to disable advertisement of 3GPP rate adaptation to all media players. Disabling this feature causes Helix Proxy to ignore any 3GPP and NADU feedback.

**Tip:** RealNetworks recommends that you keep both pull-down menus set at Yes unless you have a specific reason to disable one of the rate adaptation methods.
3. The **User Agent Settings Name** field lists all of the user agent settings stored in the Helix Proxy configuration file or the separate user agent files. Select a profile name to edit that profile. To create a new profile, click the “+” icon, and type a profile name in the **Add/Edit Profile Name** field.
4. Define which user agent strings the profile matches by clicking the “+” icon under **User Agent String Match**. Then, enter the string in the **Add/Edit String Match** field. You can create any number of user-agent strings that the profile matches.

**For More Information:** See “User Agent Strings” on page 67.

5. Under **Save Location**, click a radio button to save the profile as a separate user-agent file (recommended), or as part of the Helix Proxy configuration

file. The user-agent files are stored in the ClientProfiles subdirectory of the main Helix Proxy installation directory.

**Tip:** If you save the profile to a UAS file, the profile name you created in Step 3 is used as the .uas file name.

**For More Information:** See “User Agent Profiles” on page 64.

6. From the **Use Server-Side Rate Adaptation** pull-down menu, choose one of the following values to determine how Helix Proxy uses rate control with on-demand clips:
  - Client Select – Helix Proxy uses rate adaptation only if the media player reports that it supports one of the advertised rate adaptation methods (Helix or 3GPP).
  - Always – Helix Proxy uses rate adaptation regardless of the client capabilities. It bases rate adaptation choices on receiver reports (if sent), X-WAP values, and configuration values.
  - Never – Server-side rate control disabled for this profile. Helix Proxy will not initiate any rate shifting.
7. Using the **Use Server-Side Rate Adaptation For Live** pull-down menu, choose one of the following values to control how Helix Proxy uses rate control with live broadcasts:
  - Client Select – Helix Proxy uses rate adaptation only if the media player reports that it supports one of the advertised rate adaptation methods (Helix or 3GPP).
  - Never – Server-side rate control disabled for this profile. Helix Proxy will not initiate any rate shifting.
8. If you want this profile to inherit settings from an existing profile, select the parent profile in the **Base Profile** pull-down menu.
 

**For More Information:** See “Profile Value Inheritance” on page 66.
9. The three choices available under **Apply Presets** set profile values that determine how a media player responds to bandwidth changes. Selecting one of the preset values (Conservative, Moderate, or Aggressive) may be useful when you create a new profile. After you choose a preset, you can modify values such as the upshift and downshift depths as needed.

**Note:** Do **not** choose a preset value if you have selected a parent profile under **Base Profile**.

**For More Information:** For details about the preset values, refer to “Using the Default Profile or Preset Values” on page 94.

10. The **Default Media Rate** setting is optional. When the selected stream is encoded at multiple bandwidths, this information helps Helix Proxy to choose the initial streaming rate. If you want to set this value, enter the default rate in bits per second. This value must be smaller than the **Max Send Rate** value.

**For More Information:** See “Initial Rate Selection” on page 71.

11. Four fields allow you to set the upshift depth and downshift depth:
  - a. The value in the **Upshift Depth** field sets the upshift depth if the media player **does not** report a Target-Time value.
  - b. The value in the **Downshift Depth** field sets the downshift depth if the media player **does not** report a Target-Time value.
  - c. The value in the **Target-Time Upshift Depth** field sets the upshift depth if the media player **does** report a Target-Time value.
  - d. The value in the **Target-Time Downshift Depth** sets the downshift depth if the media player **does** report a Target-Time value.

After you enter a value, use the pull-down menu to indicate that the value is one of the following:

- percentage – a percentage of the media preroll or Target-Time value
- milliseconds – a specific number of milliseconds of buffered media

**For More Information:** See “Downshift Depth Settings” on page 80 and “Upshift Depth Settings” on page 83.

12. The value for the **Stream Switch Delay** field sets the minimum time in milliseconds after an upshift or downshift occurs that another upshift or downshift can happen. The default value is 1000 milliseconds (one second). RealNetworks does not recommend setting this value lower.
13. For **Max Send Rate**, you can select a top streaming bit rate for this profile in bits per second. For example, if you do not want a media player using this profile to receive more than a 64 Kbps stream, enter 64000 in the field.

**Tip:** If capping the sending rate is not important, leave this field set to an arbitrarily high number such as 10000000.

**For More Information:** See “Maximum Sending Rate” on page 70.

14. The **Client Capabilities** section holds values specific to each media player. You can choose to set these values yourself, or retrieve them from the player’s RDF file.

**For More Information:** See “RDF Files” on page 65 for background on RDF.

- a. A value in the **Video Predecode Buffer Size** field signals that the optional video buffering requirements defined in Annex-G are supported. The following values are accepted:

0 – Annex-G is not supported.

1 – Annex-G is supported.

$n$  – A value equal to or greater than the default buffer size defined in Annex-G means that Annex-G is supported. It also sets the buffer size to the specified number of bytes.

If the player’s RDF file contains a different value for this variable, Helix Proxy updates the UAS file to use the RDF file value.

**For More Information:** See section 5.2.3.2.1 of the 3GPP specification 26.234.

- b. If the media player supports Annex-G, the value in the **Video Decode Byte Rate** field defines the peak decoding byte rate the client is able to support. If the player’s RDF file contains a different value for this variable, Helix Proxy updates the UAS file to use the RDF file value.

**Tip:** This value is usually 8000 or higher.

**For More Information:** Refer to section 5.2.3.2.1 of the 3GPP specification 26.234.

- c. For **Retrieve X-WAP Profiles**, select Yes to enable Helix Proxy to retrieve RDF information if the client’s `x-wap-profile` header supplies a URL to the RDF file. In this case, Helix Proxy retrieves the remote RDF file

and uses it to update the UAS file information specified in the **Video Predecode Buffer Size** and **VideoDecode Byte Rate** fields.

**Tip:** The most likely reason to disable this feature is if Helix Proxy does not have an HTTP connection to the Internet, and therefore cannot retrieve the X-WAP information.

15. When you have finished defining a profile, click **Apply**.

## Using the Default Profile or Preset Values

The Helix Proxy configuration file defines a general-purpose profile named Default. Helix Proxy uses this profile if a media player's user-agent string does not match any other profiles. When you create a profile, you can have it inherit rate control variables from the Default profile.

Helix Administrator also provides three presets named Conservative, Moderate, and Aggressive. You can choose one of these presets when creating a new profile. This configures the profile with a set of values that affect the rate-shifting characteristics:

- The Conservative preset responds slowly to increases in bandwidth, upshifting only when it's highly likely that the upshifted rate will be sustainable. It also limits each upshift to a 150 percent increase over the current bandwidth. This preset downshifts quickly in response to decreases in bandwidth, however. This helps to ensure that the media player's buffer does not drain and possibly stall the presentation.
- The Aggressive preset responds quickly to increases in bandwidth, attempting to maximize the presentation quality as fast as possible when it detects a sustainable bandwidth increase. On a single upshift, it can jump to a stream that is up to 400 percent faster than the current rate. This preset reacts more slowly to bandwidth drops, though, attempting to keep the stream at its current rate as long as possible before downshifting.
- The Moderate preset provides a midpoint between the Conservative and Aggressive values. The values it sets are similar to those found in the Default profile.

The following table lists the main profile values set in the Default profile, and configured for the Conservative, Moderate, and Aggressive presets.

<b>Default Profile and Presets Values</b>					
Variable	Default	Conservative	Moderate	Aggressive	Reference
UpshiftDepth	400%	500%	400%	200%	page 83
DownshiftDepth	80%	90%	80%	50%	page 80
TargetTime UpshiftDepth	400%	500%	400%	200%	page 83
TargetTime DownshiftDepth	80%	90%	80%	50%	page 80
MinPreroll	(none)	5000 ms	2000 ms	(none)	page 77
StreamSwitchDelay	1000 ms	10000 ms	4000 ms	1000 ms	page 81
RTCPRRate	2%	3000 bps	4000 bps	5000 bps	page 73
RTCPRRate	1%	2000 bps	3000 bps	4000 bps	page 74
MaxSendRate (bps)	10000000	10000000	10000000	10000000	page 70
InitialOversendRate	400%	100%	200%	400%	page 72
MaxOversendRate	500%	150%	220%	400%	page 64
FeedbackTimeout	10000 ms	10000 ms	10000 ms	10000 ms	page 74
DownshiftOn FeedbackTimeout	1 (enabled by default)	1 (enabled by default)	1 (enabled by default)	1 (enabled by default)	page 74
EnableClient ReportedDownshift	1 (enabled by default)	1 (enabled by default)	1 (enabled by default)	1 (enabled by default)	page 82
EnableClient ReportedUpshift	1 (enabled by default)	1 (enabled by default)	1 (enabled by default)	1 (enabled by default)	page 85
UDPCongestion ControlType	tfr	tfr	tfr	tfr	page 88

### Preset BCC Settings

The Conservative, Moderate, and Aggressive presets also set values for the BCC mechanism, as shown in the following table.

<b>Preset BCC Settings</b>			
BCC Variable	Conservative	Moderate	Aggressive
DecreaseCoefficient	8	8	8
DecreaseExponent	0.5	0.5	0.5

(Table Page 1 of 2)

Preset BCC Settings (continued)

BCC Variable	Conservative	Moderate	Aggressive
DecreaseThreshold	20	4	10
IncreaseCoefficient	512000	512000	512000
IncreaseExponent	0.5	0.5	0.5
IncreaseThreshold	20	2	0
RTTUseIIR	1	0	0
TargetRatio	4	3	1

(Table Page 2 of 2)

## Enabling BCC

By default, TFRC is used for congestion control in all profiles created by Helix Administrator. To enable the use of BCC, you must modify the profile manually after you have created it with Helix Administrator.

► To enable BCC for a profile:

1. Depending on where you created the profile, open the profile's .uas file or the Helix Proxy configuration file in any text editor.

2. Search for the profile's RateControl sublist:

```
<List Name="RateControl">
```

3. Check for the presence of the UDPCongestionControlType variable in this list.

- a. If the UDPCongestionControlType variable is present, set the value to BCC.

- b. If the UDPCongestionControlType variable is absent, add the variable with a value of BCC as shown here:

```
<List Name="RateControl">
```

```
...other rate control variables...
```

```
<Var UDPCongestionControlType="BCC"/>
```

```
</List>
```

4. Save and close the file.

## PROXY ROUTING

One Helix Proxy can request content by way of another Helix Proxy rather than directly from a server. This can benefit security and traffic flow by handling Internet-bound requests through strict rules and a single access point to the Internet.

**Tip:** Proxy routing is designed for large enterprises in which subnet traffic is routed through proxy software. It is not recommended for use in other scenarios, as the increased latency and administrative overhead are appropriate only to controlled network situations.

**Note:** The redundant proxies feature provides failover protection from one proxy to another. For more information on this feature, refer to the section “Setting Up Proxy Redundancy” on page 54.

## Understanding Proxy Routing

Proxy routing allows you to route certain requests to various Helix Proxies through another Helix Proxy. Uses for this feature include routing all requests for locally-served material directly to an intranet-based Helix Server, while forwarding all other Internet-based requests through a gateway Helix Proxy.

Proxy routing is also known as *chaining* or *parent/child* proxies. Using the latter terminology, the main Helix Proxy that handles requests bound for the Internet is called the *parent* proxy, and is typically the only proxy that receives media over the Internet. Media players generally send requests to a *child* Helix Proxy on a local subnet. The child proxy routes the request through the parent proxy, then streams the content to the media player.

**Note:** Although it is possible for a child Helix Proxy also to act as a parent Helix Proxy, RealNetworks does not recommend this configuration due to the compounding of network and

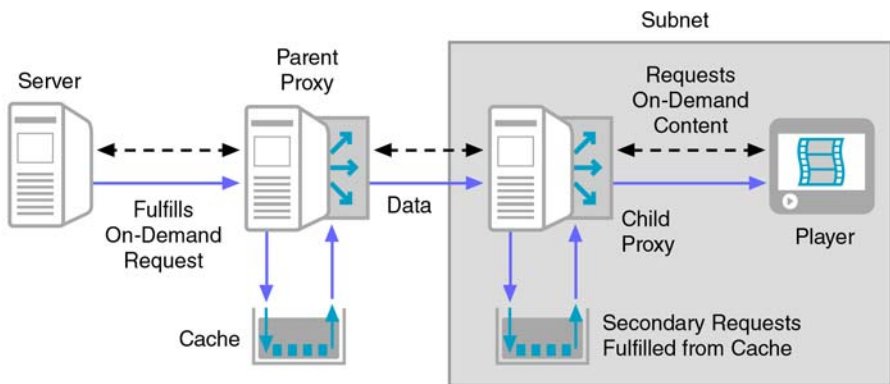
application latency. In other words, all proxies on a subnet should be child proxies to one or more parent proxies.

## Caching and Proxy Routing

Caching, which the section “Content Caching” on page 19 explains, is the standard method that Helix Proxy uses to obtain and stream on-demand content to media players. When proxy routing is used, a media player requests a clip through the child Helix Proxy, which forwards the request to the parent proxy. The parent then sends the request to the origin Helix Server.

The stream data travels back from the origin server to the parent proxy, then to the child proxy, and finally to the media player. Both the parent proxy and the child proxy cache the clip data, so that they can fulfill later requests for the same clip more quickly. Even when the child proxy has the requested clip cached, subsequent requests are routed through the parent to the origin server to ensure that the cached material is not outdated. If the cached content is current, the child streams the content directly from the cache, eliminating the data chain between the origin server and the child proxy.

### Caching Used with Helix Proxy Routing



**Note:** A parent Helix Proxy can also stream content to media players while simultaneously streaming data to a child Helix Proxy.

## Pull-Splitting with Proxy Routing

Pull-splitting is the standard method for delivering a live stream, as described in “Pull-Splitting” on page 21. When a media player requests a live stream

through a child Helix Proxy, the child forwards the request to the parent proxy, which then contacts the origin Helix Server. The server sends the data to the parent Helix Proxy, which transmits it to the child Helix Proxy.

When the child Helix Proxy receives the stream, it splits the stream to all players requesting the stream. Stream splitting does not occur automatically at the parent proxy, however. If a media player connects directly to the parent and requests the same live stream, the parent proxy sends a separate request to the origin server. In this case, both the child and parent proxies maintain open control channels to the origin server, with the child proxy's channel passing through the parent proxy. Likewise, both the child and the parent proxy receive a separate data stream, with the child's stream also passing through the parent proxy.

### **Pass-Through Delivery with Proxy Routing**

When content cannot be cached or split, Helix Proxy uses the pass-through feature, which the section “Pass-Through Delivery” on page 22 describes. Proxy routing with pass-through delivery works much the same as it does with caching or splitting, except that the media is never cached or split, so each request must be fulfilled by the origin Helix Server. All content streams from the origin server, passing through both the parent and the child proxies.

**Note:** The parent Helix Proxy always maintains the control connection to the origin Helix Server. The child Helix Proxy never contacts the origin Helix Server directly.

### **Authentication with Proxy Routing**

If you implement authentication on all proxies as described in Chapter 11, both the parent and the child proxy prompt each user to enter a valid user name and password. Each proxy therefore needs to define each user in its authentication database. For each proxy, the user name and password combination for a given user can be different, but keeping the same user name and password for each user across all proxies results in simpler administration and a better user experience.

## Setting Up Proxy Routing

To set up proxy routing, you create routing rules on each child Helix Proxy. These rules determine which requests are forwarded to a parent Helix Proxy, and which requests are handled normally by the child Helix Proxy.

### Using Proxy Routing Rules

A routing rule defines a URL or class of URLs, and instructs Helix Proxy how to request the content for that URL. Within a rule, an asterisk (\*) indicates a wildcard section for a URL. The following are examples of rule URLs that use wildcards, listed from more specific rules to more general rules:

```
helixproxy.department.example.*  
helixproxy.example.*  
helixproxy.*.com  
*.example.com  
*.com  
*
```

Each rule indicates whether, for the specified URL class, a parent proxy is used, or the child handles the URL normally. The last rule in the preceding list of examples simply uses a wildcard for the URL. This forwards all requested URLs to a specified parent Helix Proxy.

### Wildcard Restrictions

Note the following restrictions when using wildcards:

1. You can use only one asterisk per rule. For example, the following rule is valid:

```
*.example.com
```

But the next rule is **not** valid:

```
*.example.*
```

2. The asterisk cannot be used with a string. It can be used only within periods. For example, the following rule is not valid:

```
real*.example.com
```

### Rule Examples

To forward all requests from a child Helix Proxy to a parent proxy, you define a single rule using just a wildcard (\*), and specify the parent proxy as the URL recipient. You can create more specific rules if needed, however. Suppose that

you want the proxy to route all intranet-based media streams to an internal Helix Server behind your firewall, while shuttling all Internet-based media traffic through a parent proxy. You would define rules such as the following:

```
*.example.com    route to local Helix Server
*                route to parent Helix Proxy
```

Assuming that your intranet uses the example.com address, the first rule intercepts all internal media requests, routing them to your internal origin server. The second rule captures all other media requests, routing them through the parent proxy.

## Defining Proxy Routing Rules

When setting up proxy routing, you define any number of routing rules on the child Helix Proxies. You do not need to make changes to the parent Helix Proxy.

### ► To set up a proxy routing rule:

1. In Helix Administrator, click **Proxy Setup>Proxy Routing**.
2. In the **Routing Rules** area, click the “+” icon.
3. In the **Edit Rule Description** field, type a description for the rule. This is for your reference only, and does not affect the routing feature.
4. In the **Rule URL** field, define the URL class, using wildcards (“\*”) as appropriate, that this rule intercepts. Do not include protocol designations, such as rtsp://, or any mount points or directory listings. For more information, see “Using Proxy Routing Rules” on page 100.
5. If the rule should redirect requests to a parent proxy, enter the host name or IP address of the parent Helix Proxy in the **Parent Name** box.
6. To forward URLs to the parent proxy, choose **Yes** in the **Use Parent Proxy** pull-down list. If you choose **No**, the child proxy handles the URLs intercepted by the rule normally.
7. If you designate a parent proxy, you need to specify the RTSP and MEI ports on that Helix Proxy:
  - The **Parent RTSP Port** value specifies the port the parent proxy uses to listen for RTSP requests. This is usually 554.

- In the **Parent MEI Port** box, specify the port on the parent proxy to which the child proxy directs cache requests. This is usually 7878.
8. Helix Proxy examines rules in order, applying the first viable rule it finds. You should therefore place specific rules ahead of general rules. For example, a rule for news.example.com should precede a rule for \*.example.com. Set the order of a rule in the **Routing Rules** box by highlighting the rule and clicking the up or down arrow icon.
  9. Click **Apply**.

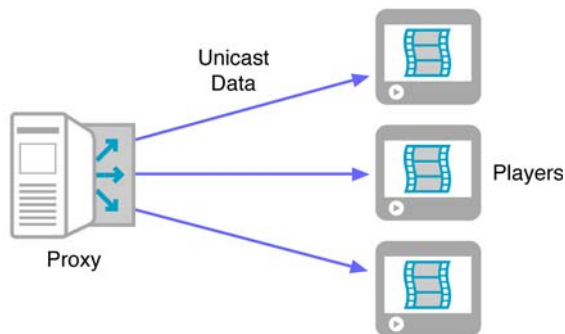
## MULTICASTS

Multicasting helps you conserve bandwidth during a live or simulated live broadcast. Although it can increase the audience for an event by reducing the broadcasting bandwidth, it requires a specially configured network, and is more suited for intranets than Internet delivery.

### Understanding Multicasts

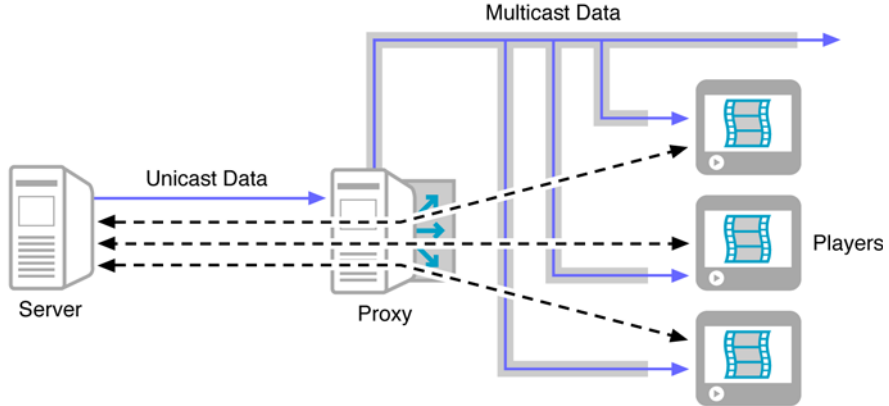
In a default broadcast, called a *unicast*, Helix Proxy pulls a single stream from Helix Server, then splits the stream to each media player (the standard pull-splitting feature) The following figure illustrates a unicast.

#### Unicasting



Multicasting enhances pull-splitting by sending a single data stream to multiple media players, rather than a separate stream to each player. As shown in the following illustration, media players establish a data connection to the stream, rather than to Helix Proxy. This greatly reduces the amount of data broadcast on the network between Helix Proxy and the media players.

## Multicasting



Helix Proxy uses a form of multicasting called *back-channel multicasting*. In addition to the connection to the data stream, each media player maintains a control channel to Helix Server through Helix Proxy, as shown by the dashed lines in the preceding illustration. The control channel provides several features:

- The player uses the control channel to send commands such as **Stop**. Because each player uses a control channel, back-channel multicasting is limited to the number of media player connections licensed to your Helix Proxy and Helix Server.
- The channel allows Helix Proxy and Helix Server to receive a user name and password when authentication is used.
- The channel enables the Proxy Monitor described in Chapter 14 to track how many players are viewing the multicast.

**Note:** Helix Proxy does not support scalable multicasting, which does not use a back-channel.

## Media Players and Formats for Multicasting

Multicasting generally uses RTSP to send control information over a TCP channel and live broadcast data over a UDP channel. The following table

summarizes the media players, control protocols, and media types that you can multicast.

#### Media Players and Multicast Media Types

Media Player	RTSP with RealMedia	RTSP with MPEG	RTSP with QuickTime	MMS with Windows Media
RealPlayer 5 and earlier	No	No	No	No
RealPlayer G2 and later	<b>Yes</b>	<b>Yes</b>	No	No
Windows Media Player	No	No	No	No
QuickTime Player	No	<b>Yes</b>	<b>Yes</b>	No
RTP-based player	No	<b>Yes</b>	No	No

#### Unicast Failovers

If a RealNetworks media player is not multicast-enabled, or cannot connect to the back-channel multicast, it fails over to a unicast automatically. This ensures that all players can receive the broadcast. Because each unicast stream consumes extra bandwidth and Helix Proxy overhead, you can choose to disable the failover feature and provide just the multicast. In this case, a player not able to participate in a multicast receives an error message when attempting to connect to the broadcast.

**Tip:** You can use back-channel multicasting with the failover feature for all broadcasts to RealNetworks media players. By default, players attempt a back-channel multicast connection first, switching to unicast if the failover feature is enabled, and the multicast is not available. Hence, enabling an automatic multicast for all broadcasts can help conserve bandwidth.

## Configuring a Network for Multicasts

To use multicasting, Helix Proxy, media players, routers, switches, and all other networking devices between them must be multicast-enabled. For this reason, multicasting is primarily used on intranets. However, it is possible to deliver multicasts over the Internet where intermediary network devices have been multicast-enabled. Before using multicasting, verify the following with your network administrator:

- Routers and all equipment in your network are multicast-enabled.

- The machine running Helix Proxy is correctly configured for multicast support.

**Tip:** RealNetworks media players are configured for multicast by default, although viewers can turn off multicast support in their player preferences.

## Multicast Addresses

A multicast requires the use of a continuous range of multicast addresses on your network. Valid ranges are between 224.0.0.0 and 239.255.255.255. Check with your network administrator about which multicast addresses are available on your network. On the public Internet, certain ranges in the multicast address space (from 224.0.0.0 to 224.0.0.255) are reserved, and cannot be used.

**Note:** Helix Proxy does not support multicasting to IPv6 addresses.

**For More Information:** See “Assigned Numbers,” RFC 1700, available at <http://www.ietf.org/rfc/rfc1700.txt>.

## Packet Time to Live

All multicast broadcasts include a “time to live” feature. As a multicast data packet passes through a multicast-enabled router, its time to live decreases by 1. When the value reaches 0, the router discards the data packet. When you set up a multicast, you specify a time to live of 0 to 255. The larger the value, the greater the distance a packet can travel. The default value of 16 typically keeps multicast packets within an internal network. The following table summarizes possible values.

**Time to Live (TTL) Values**

TTL Value	Packet Range
0	local host
1	local network (subnet)
16	intranet
32	site
64	region

(Table Page 1 of 2)

**Time to Live (TTL) Values (continued)**

TTL Value	Packet Range
128	continent
255	world

(Table Page 2 of 2)

**Multicasts with Multiple Network Interface Cards**

If your Helix Proxy machine has multiple network interface cards (NICs), and you want to ensure that Helix Proxy always uses a particular NIC for multicasts, use your operating system to set a default address. On Windows, set IP bindings as described in “Binding To An IP Address” on page 48. On UNIX, use the **route** command to associate the multicast route with the appropriate NIC.

**Address Requirements**

Determining the number of addresses you need for a back-channel multicast is straightforward. You need just one address per bit rate, regardless of the number of streams. So although a single-rate video technically delivers two streams, one for the audio track and one for the visuals, both tracks can use the same address.

**SureStream Broadcasts**

For SureStream, you need to reserve one address for each bit rate encoded into the stream. If a SureStream stream is encoded for three audience bandwidth targets, for example, you need three addresses. The duress streams encoded for a particular bandwidth target are not used, and do not require additional addresses.

**Note:** Media players cannot shift between SureStream streams during the multicast.

**Automatic Multicasts**

If you intend to leave back-channel multicasting on for all broadcasts, you need enough multicast addresses to accommodate your typical broadcast. It’s a good idea to implement a policy, such as using only two bandwidth targets when broadcasting RealVideo. If you have not reserved enough addresses for a particular multicast, the event is unicast automatically, as long as you have not disabled the failover feature.

## Configuring Back-Channel Multicasting

The following procedure describes how to set up Helix Proxy for back-channel multicasting. Minimally, you need to define your multicast address range. Other features are optional.

► To set up back-channel multicasting:

1. In Helix Administrator, click **Proxy Setup>Multicasting**.
2. The **Enable Multicast** list turns on this feature. Ensure that the default value of **Yes** is selected. If you set this to **No**, multicasting is disabled, and all media players use unicasting for all broadcasts.
3. Set the **Enable SAP** list to **Yes** to announce the multicast session.
4. In the **RTSP Port** box, list the port number on media players where Helix Proxy directs RTSP multicast streams. The default is 3554.
5. Specify the range of addresses to which you want to multicast streams by filling in the **IP Address Range** box. Helix Proxy uses the first available addresses in this range. See “Address Requirements” on page 107 for information about the number of addresses you’ll need.
6. Indicate how far multicast packets can travel over a network by typing a value in the **Time to Live** box. For more information, see “Packet Time to Live” on page 106.
7. To allow missing packets to be resent to media players that request them, select **True** from the **Resend** list. Resending packets adds network overhead, but delivers higher-quality multicasts.
8. If you want to deliver the broadcast through multicasting alone, choose **Yes** from the **Multicast Delivery Only** list. Media players that cannot use multicasting will not be able to connect to the broadcast. Use this feature when broadcasting only to multicast-enabled media players, or if you are multicasting a high-bandwidth presentation and do not want to provide a unicast option.

**Warning!** Selecting this option prevents you from unicasting any broadcast to RealNetworks media players. If you want to reuse unicasting after your multicast, turn this option off when the multicast ends.

9. The **Access Rules** section lets you restrict the range of media players that can connect to the multicast. A predefined rule allows all multicast-enabled players with access to the broadcast URL to connect to the multicast. You can delete this rule, modify it, and set up other rules as necessary:
  - a. To add a new rule, click the “+” icon and, optionally, change the default name in the **Edit Client Access Rule Description** box. Because the access rules simply list addresses of media players that have access, the order of the rules in the **Access Rules** box does not matter.
  - b. For the highlighted rule, enter an IP address or domain name for acceptable players in the **Client IP Address or Hostname**. The value Any is predefined for the first rule, meaning that all IP addresses are accepted. To restrict the range, delete the rule, or change its value and set a netmask.
  - c. In **Client Netmask**, specify the range of media player IP addresses around the one you entered in the preceding step by selecting a bit mask. If **Client IP Address** is set to Any, though, leave **Client Netmask** set to None. See Appendix B for details about assigning a range of IP addresses using a bit mask.
10. Click **Apply**.

**Tip:** General access control rules, which are described in Chapter 10, are enforced before multicast access rules. A media player excluded by general access control will not connect to any multicast, regardless of the multicast access rules.



## MEDIA PLAYER CONFIGURATION

For media players such as RealPlayer to use Helix Proxy, you either configure the media players individually, or configure your network to route streaming media requests to the proxy. This chapter describes how to set up RealPlayer and Windows Media Player to contact Helix Proxy using either method.

### Manual and Automatic Configuration

Most media players, such as RealPlayer and Windows Media Player, contain an option to contact a proxy rather than to send requests directly to Helix Server. In the player preferences, the user enters the IP address (or host name) and port number of the proxy. If you choose to connect media players to your Helix Proxy this way, you must either set up the media players yourself, or send instructions to the users about how to configure their players.

The second option is to configure Helix Proxy to intercept media requests. This does not require any special media player configuration. However, it does require the use of software or hardware that routes TCP traffic by destination port, such as a Layer-4 switch.

### Configuring RealPlayers Manually

If you choose to configure RealPlayers to connect directly to Helix Proxy, follow the procedure below. These instructions are specific to version 3 of RealPlayer. The preference dialog for earlier versions of RealPlayer may differ.

► **To configure RealPlayer:**

1. In RealPlayer, select **Tools>Preferences**.
2. Open the **Connection** category and select **Proxy**.
3. Under the **Streaming Settings** section, click the **Change Settings** button.

4. In the **RTSP Proxies** section, click **Use Proxies**.
5. In the **RTSP** field, type the IP address or host name of your Helix Proxy.
6. In the **Port** fields, type the number of the Helix Proxy port to which this media player should send its RTSP requests (typically 554).

**For More Information:** The section “Defining Communications Ports” on page 45 explains how to set the port numbers on Helix Proxy.

7. Click **OK**.

## Configuring Windows Media Players Manually

The following sections explain how to configure Windows Media Player to work with Helix Proxy. Note that starting with version 11, Windows Media Player no longer supports the MMS protocol.

### Windows Media Player Versions 7 through 10

Follow the instructions below to configure Windows Media Players version 7.1 through version 10 to connect directly to Helix Proxy.

► **To configure Windows Media Player version 7 through 10:**

1. In Windows Media Player, select **Tools>Options**.
2. Select the **Network** tab.
3. Under Proxy Settings, select **MMS**.
4. Click the **Configure...** button.
5. Select the **Use the following proxy server** radio button.
6. In the **Address:** text box, type the IP address or host name of your Helix Proxy computer.
7. In the **Port** box, type the number of the Helix Proxy port number to which this media player should send its MMS requests (usually 1755).

**For More Information:** The section “Defining Communications Ports” on page 45 explains how to set the port numbers on Helix Proxy.

8. Click **OK**.

## Windows Media Player Version 11 and Later

Windows Media Player 11 does not request media using the MMS protocol. When it encounters an MMS URL for on-demand or live content on Helix Server, Windows Media Player rejects the MMS URL and attempts to contact Helix Server over HTTP. If the request is successful, Helix Server streams the clip as a Windows Media stream cloaked as HTTP. If that connection is unsuccessful, the player attempts to request the content over RTSP.

Windows Media Player 11 does not provide an option to use an MMS proxy. Instead, its player preferences contain options to use an HTTP proxy and an RTSP proxy:

- For the HTTP proxy, you can select any HTTP proxy available to you. You cannot use Helix Proxy, however, because Helix Proxy does not proxy any media using HTTP.
- For the RTSP proxy option, you can specify Helix Proxy. Note the following, however:
  - Because Helix Server does not support Windows Media over RTSP, Helix Proxy cannot proxy any Windows Media content residing on Helix Server for Windows Media Player 11. Streams originating from Helix Server can be delivered only by an HTTP proxy as HTTP-cloaked Windows Media.
  - Helix Proxy can proxy on-demand and live, RTSP-based Windows Media streams originating from a Windows Media Server. However, all streams are delivered in pass-through mode only. Helix Proxy does not cache on-demand clips or split live streams.

## Configuring Automatic Proxy Redirection

If your network uses a Layer-4 switch or similar mechanism, you can typically configure it to route RTSP requests for media residing on external Helix Servers to your local Helix Proxy. This redirection method applies only to RTSP requests, and it assumes that the switch is situated to intercept *all* media player requests before they reach the origin Helix Server.

The redirection method uses a plug-in named `redirect.so.6.0` on UNIX or `redi3260.dll` on Windows. This plug-in ships with Helix Proxy, but is disabled by default. When configured and enabled, this plug-in issues an RTSP redirection command to the media player.

The following steps explain the sequence of events involved in automatic redirection:

1. A media player requests a clip from Helix Server, using a URL such as `rtsp://helixserver.example.com/music.rm`.
2. The local network's Layer-4 switch, which monitors all client traffic, intercepts TCP packets destined for the RTSP port (typically 554) on the external host. By overwriting the packets' destination IP address and, if necessary, the port number, it sends the packets to the listen port of the Helix Proxy redirector plug-in.
3. On its listen port, the Helix Proxy redirector plug-in receives the TCP packets containing the media request.
4. Helix Proxy issues the media player an RTSP option 305, which directs the player to use a proxy. It also provides the media player with the Helix Proxy address and RTSP listen port.
5. When the media player receives the 305 option response, it tears down its connection to the external Helix Server. It then opens a new TCP connection to the Helix Proxy RTSP listen port.
6. Helix Proxy delivers the requested media to the media player.

## Media Player Requirements

To use automatic redirection, you do **not** configure media players manually to use a proxy. There are no specific configuration requirements for RTSP-based media players. However, media players **must** be capable of accepting an RTSP option 305. All versions of RealPlayer released since 1998 support this option.

**For More Information:** The RTSP RFC, available at <http://www.ietf.org/rfc/rfc2326.txt>, defines the RTSP option codes.

## Network Switch Configuration

To implement automatic redirection, your network's Layer-4 switch (or equivalent mechanism) must intercept and overwrite outgoing packets for media player requests. Consult the documentation for your switch for specific configuration instructions. In general, the switch must do the following:

- It must support port-based switching of TCP traffic destined for the RTSP port of external servers. This is typically port 554.
- When re-routing a TCP packet, the switch must overwrite the packet's IP header, changing the external server address to the Helix Proxy host address.
- If the redirector plug-in does not use the standard RTSP port (554) as its listen port, the switch must also change the port value in the request URL. This directs the packet to the Helix Proxy redirector plug-in's listen port.
- The switch must **not** intercept TCP-based media player traffic that has been successfully redirected to the Helix Proxy host address and RTSP listen port.

## Helix Proxy Configuration

You configure the Helix Proxy redirector plug-in by editing the proxy configuration file. You define a redirector listen port that the plug-in uses to receive packets re-routed by the switch. If necessary, you also change the Helix Proxy RTSP port.

► **To configure Helix Proxy for automatic redirection:**

1. Using any text editor, open the Helix Proxy configuration file (`rmproxy.cfg`). This file resides in the Helix Proxy main installation directory.

**For More Information:** For more about the configuration file, refer to Appendix A.

2. Add the following RTSP redirector list and variables to the configuration file. This list should not be contained within any other list:

```
<List Name="RTSPRedirector">
  <Var Port="Plugin_Listen_Port"/>
  <Var RedirectToAddress="Proxy_Address"/>
  <Var RedirectToPort="RTSP_Listen_Port"/>
</List>
```

Use the following variable values:

Port	The port that the plug-in uses to listen for redirected RTSP requests. This is commonly port 554. However, it can be any open port. If port 554 is not used, the switching mechanism must overwrite TCP packets in media requests to use the selected port number. If port 554 is used, Helix Proxy cannot use that port as its RTSP listen port.
RedirectToAddress	The IP address of Helix Proxy. The proxy instructs media players to connect to it using this address.
RedirectToPort	The port Helix Proxy uses to listen for RTSP requests. The proxy instructs media players to send requests to this port.

For example:

```
<List Name="RTSPRedirector">  
  <Var Port="554"/>  
  <Var RedirectToAddress="127.118.32.0"/>  
  <Var RedirectToPort="1091"/>  
</List>
```

3. In the configuration file, find the following string:

```
<Var RTSPPort="value"/>
```

Change the RTSPPort value to the same value used by the RedirectToPort variable. For example:

```
<Var RTSPPort="1091"/>
```

**Note:** If your switching mechanism **cannot** recognize and make exceptions for requests sent to Helix Proxy on port 554, Helix Proxy **cannot** use the default 554 as its RTSP listen port.

4. Save and close the configuration file.
5. Restart Helix Proxy.



## SECURITY

The administrator must ensure security for the network where Helix Proxy resides, as well as consider the needs of media clients that may be behind restrictive firewalls. The following chapters help you handle these security issues.



## FIREWALLS

Firewalls may present communications problems to Helix Proxy. This chapter helps you to become familiar with network firewalls to help you use Helix Proxy successfully. It first provides background on firewalls and network protocols. It then recommends ways to work with firewalls to give viewers the best possible streaming media experience. Finally, it lists the communications ports that RealNetworks components use.

### How Firewalls Work

A firewall is a software program or device that monitors, and sometimes controls, all transmissions between an organization's internal network and the Internet. However large the network, a firewall is typically deployed on the network's edge to prevent inappropriate access to data behind the firewall. The firewall ensures that all communication in both directions conforms to an organization's security policy.

Firewall technologies are configurable. You can limit communication by direction, IP address, protocol, ports, or numerous other combinations. Firewalls positioned between your Helix Proxy and other computers may cause communication failures if the firewall does not allow for the types of communication Helix Proxy requires. These other computers may be media players or servers set up as origin transmitters.

If you have access to the firewall, you can configure it to enable the ports, protocols, and addresses that optimize Helix Proxy communication. In some cases, however, your organization's security policy may prevent optimal streaming. For example, firewalls configured to only allow TCP traffic may cause the user to see frequent buffering of clips. User experience of the presentation is compromised; greater latency and startup times affect the time needed to view the clip, and delivery of the clip requires more total bandwidth.

## Protocol Layers

A protocol is a language that computers use when communicating over a network. The Transmission Control Protocol/Internet Protocol—commonly called TCP/IP—encompasses a suite of protocols upon which the Internet is built. TCP/IP protocols work on a layering principal, in which each layer is assigned a specific network task.

For communication to occur, a source computer sends a message from its highest network layer to its lowest. The lowest network layer at the source forwards the message over the network. When the message arrives at the destination computer, it must pass through the exact same layers, but in reverse order.

Each network layer uses specific protocols to perform its task. Packets passed down from upper layers are tucked inside lower layer packets. This is called *encapsulation*. By encapsulating packets, a layer can handle its responsibilities without understanding the preceding layer. Through this layering scheme, a destination layer on one computer receives exactly the same object sent by the corresponding source layer on another computer.

For example, an application such as a Web browser packages data, such as a Web page request made over HTTP, at the application layer, passing it to the lower transport layer. There, the HTTP request packets are bundled into TCP packets that are then delivered to destination Web server. When the Web server receives the source TCP packets, it strips off the TCP shells, and bumps the HTTP message up to the destination computer's application layer. This layer, in turn, delivers to the HTTP-based request to the Web server.

**Note:** Network layering is a complex topic. This section omits discussion of additional layers required to deliver packets over a network, focusing instead on the transport and application layers, and the protocols relevant to streaming media for each.

## Transport-Layer Protocols

All transport-layer protocols transfer data between hosts. The transport-layer protocol in use can greatly affect the quality of the stream received. There are two main transport protocols used on IP networks: TCP and UDP. Helix Proxy utilizes both of these protocols, and the choice of protocol is generally negotiated automatically by the Helix Server and media players involved.

## Transmission Control Protocol (TCP)

Helix Proxy can use TCP in a number of ways. Because TCP offers a single channel for bi-directional communication, Helix Proxy uses it as a control channel to relay commands from media players to Helix Server about passwords and user commands such as pause and fast-forward. The TCP protocol also guarantees delivery of packets, and has built-in congestion control that helps to provide reliable communication.

On the down side, TCP responds slowly to changing network conditions, and creates network overhead through its error checking facility. For this reason, TCP is best suited for delivering low-bandwidth material like passwords or user commands. In some cases, TCP can facilitate communication through a firewall. For example, firewalls that block UDP traffic between Helix Proxy and its media players may permit TCP connections.

## User Datagram Protocol (UDP)

Helix Proxy uses UDP packets to deliver data to media players on its data channel. Media players send UDP-based requests to Helix Proxy (which get relayed to Helix Server) when packets on the data channel have not arrived. Because the transport does not consume as much network overhead, it can deliver packets faster than TCP.

Because video and audio data typically consume large amounts of bandwidth, it makes the most sense to use UDP to deliver streaming media. For this reason, origin Helix Server use UDP as the default for server-to-proxy communication.

## Application-Layer Protocols

Helix Proxy uses two application-layer protocols to deliver streaming media to media players: RTSP or MMS. The following table summarizes their use.

**Application-Layer and Transport-Layer Protocols**

Player Software	Application	
	Protocol	Transport Options
RealPlayer G2 and later; QuickTime Player	RTSP	TCP and UDP, or TCP only
Windows Media Player	MMS	TCP and UDP, or TCP only

### Real-Time Streaming Protocol (RTSP)

A standards-based protocol designed for serving multimedia presentations, RTSP is useful for large-scale broadcasting. RTSP uses TCP for player control messages, and UDP for video and audio data. RTSP can also use TCP to deliver data, but this is not recommended.

### Microsoft Media Services (MMS)

The MMS protocol is designed specifically for serving multimedia presentations. Although it is not standards-based, you can use it to broadcast live or on-demand Windows Media clips to Windows Media Player. MMS uses TCP for player control messages, and UDP for video and audio data. MMS can also use TCP to deliver data.

### HyperText Transfer Protocol (HTTP)

HTTP is typically used for Web pages. With Helix Proxy, HTTP is used to display Helix Administrator pages and HTML-based documentation.

## Packet Formats

All Internet data is delivered in IP packets. But just as TCP or UDP can encapsulate a control protocol for streaming media, IP packets can encapsulate data packets in formats designed to deliver streaming media data. Helix Proxy uses the RDT and RTP packet formats—either of which can be delivered in either TCP or UDP internet protocol.

### RealNetworks Data Transport (RDT)

When Helix Proxy communicates to RealPlayer over RTSP, it uses RDT as the packet format. A proprietary format, RDT allows the use of proprietary RealMedia features such as SureStream.

### Real-Time Transport Protocol (RTP)

RTP is a standards-based packet format designed as the companion to the RTSP protocol. QuickTime Player, for example, uses RTP as its packet format. Helix Proxy fully supports RTP, and shifts to RTP automatically when streaming to an RTP-based media player such as QuickTime Player. RealPlayer also supports RTP, using this format when receiving data from RTSP/RTP servers.

## Communicating with Media Players Behind Firewalls

The following sections explain how Helix Proxy uses two connections, known as *channels*, to communicate with media players. Most media players, including RealPlayer, can work around situations in which the first communication fails because the player resides behind a firewall that blocks the preferred protocol. Typically, the player shifts the data channel to the less efficient TCP, which is less likely to be blocked than UDP.

**Tip:** Keep in mind that the ports a media player uses on Helix Proxy can vary, as described in Chapter 8.

### Control Channel

Helix Proxy and the media player first open a *control channel* to communicate. Over this channel, Helix Proxy initially requests and receives passwords, and sends information to the player about the requested media, such as the clip's name and length. Using this channel, media players can send instructions such as fast-forward, pause, and stop. The following figure illustrates this channel.

#### Control Channel Between a Server or Proxy and the Media Player

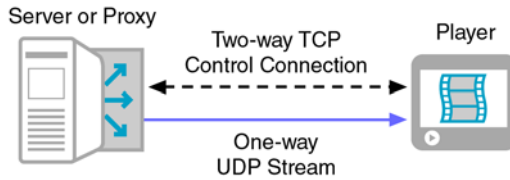


The media player attempts to open the control connection using TCP. It uses the port designated for the streaming media protocol, which is typically port 554 for the RTSP protocol. If the firewall does not allow TCP on the chosen port, the request is denied and the user receives an error message.

### Data Channel

When the TCP connection is allowed on the control channel, the media player negotiates a *data channel* for receiving media clip packets. The player tries to set up the most optimal data delivery method, which is typically the more efficient UDP transport. The following figure illustrates a one-way UDP data channel delivering media packets, while the media player maintains its control channel with the server or proxy.

### Control Channel Between a Server or Proxy and the Media Player



### On-Demand Data Channel Negotiation

When the media player requests on-demand content, it first tries to open a UDP channel. Most RealPlayers use the port range 6970 through 32000. Standards-compliant RTP-based players accept UDP communication on ports 1024 through 65535. Early versions of RealPlayer used a smaller UDP port range. If UDP is not allowed, the player requests that the data be sent over TCP on the established control channel.

### Live Data Channel Negotiation

When the request is for live content, the media player tries up to three delivery methods, starting with the most optimal:

- The media player first requests a multicast, which requires a multicast-enabled network and firewall. Multicast uses the UDP transport protocol, and the RTSP application-level protocol.
- If multicast is not available, the player requests that the material be sent using UDP on ports 6970 through 32000.
- If UDP cannot pass through the firewall, the player requests delivery using TCP on the established control channel.

**For More Information:** The section “Changing Pull-Splitting” on page 44 explains how you can set Helix Proxy to use only UDP or TCP.

## HTTP Cloaking

When it connects directly to Helix Server, RealPlayer can circumvent highly restrictive firewalls by receiving media data over HTTP. This communication method, known as *HTTP cloaking*, does not work when RealPlayer communicates through Helix Proxy, which has no option for HTTP delivery. Thus, if the firewall prohibits RTSP, Helix Proxy will not be able to proxy streams on behalf of media players.

## Firewall Configurations

The firewall that provides the best experience for media players allows streaming media by enabling TCP and UDP traffic. Several firewall vendors already include this type of streaming media support. View the RealNetworks firewall page at <http://service.real.com/firewall> to find a vendor. You can also modify your existing firewall with the help of the free RealNetworks Firewall Administrator's Proxy kit.

The next best option is a firewall that allows a TCP control channel and a TCP data channel. Your firewall administrator can easily make this change to the firewall. However, the quality of the connections will not be as good with this configuration.

### Locating Helix Proxy Near the Firewall

A realistic deployment of Helix Proxy within or near a secure network is to place it inside a network firewall or in a secure perimeter network known as the DMZ (de-militarized zone.) In such a deployment, media players typically are not allowed to access the public Internet or other non-local networks directly. Instead, players send requests to Helix Proxy, which makes and receive Internet connections outside the secure network. In this arrangement, only Helix Proxy is exposed to network traffic beyond the confines of the secure firewall.

The firewall must satisfy the following criteria:

- All media players residing within the secure network need to connect to Helix Proxy using TCP.
- Helix Proxy needs to be able to send both TCP and UDP traffic to its media players.
- If Helix Proxy contacts origin Helix Servers outside the secure network, the firewall must allow Helix Proxy to make outbound TCP connections on several ports. Additionally, Helix Proxy will need to receive UDP traffic from those remote Helix Servers.

**For More Information:** Refer to the next section, “Default Ports” on page 129, for specific information on the ports that are needed.

## Working with Multiple IP Addresses

If your firewall allows connections to Helix Server only from certain IP addresses, make sure that it permits traffic on all the addresses used by Helix Proxy. When the Helix Proxy machine has multiple IP addresses (either from the Network Interface Card or multiple virtual addresses), you use the IP bindings feature to select the addresses that Helix Proxy uses. Helix Proxy then makes its outgoing connections using the operating system's routing table.

**For More Information:** The section “Binding To An IP Address” on page 48 explains how to specify IP addresses used by Helix Proxy. Refer to your operating system's TCP/IP documentation for more information.

## Firewall Types

Firewalls can be categorized into roughly six types, though a particular firewall may combine more than one type of protection. The type of firewall in use by your organization will affect the method that Helix Proxy uses to stream content to media players. The following sections provide more details on these basic types of firewalls:

- application-level proxy
- transparent proxy
- packet filter
- stateful packet filtering
- SOCKS
- network address translation

### Application-Level Proxy Firewall

Application-level firewalls determine if a requested connection between a computer on the internal network and one on the outside is permitted. If the connection is authorized, the firewall mimics the requesting software, setting up the necessary communication links between the two computers. As an intermediary, the firewall can monitor the communication between the two networks and suppress any unauthorized activity.

Because an application-level firewall acts as an intermediary between RealPlayer and Helix Proxy (or between Helix Proxy and Helix Server), the firewall itself must know how to handle the RTSP and MMS protocols. The

user must configure the media player to contact a proxy or firewall machine, as described in Chapter 8.

### Transparent Proxy Firewall

A network administrator configures a transparent proxy firewall to intercept requests for streaming media.

### Packet Filter Firewall

Rather than impersonating an application, network-level firewalls examine the packets of information sent at the transport level to determine whether a particular packet should be blocked. Each packet is either forwarded or blocked based on a set of rules defined by the firewall administrator.

A common configuration for network-level-filtering firewalls is to allow all connections initiated by machines inside the firewall, and to restrict or prohibit all connections made by machines outside the firewall. For most programs, this works well since they usually only establish a single outbound TCP connection.

However, RealPlayer and Helix Proxy (or Helix Proxy and Helix Server) maintain a TCP control channel and a UDP data channel. The TCP control channel initiated by the media player will pass through a packet filter firewall. But because network-level filters block incoming UDP, the UDP stream sent by Helix Proxy will be blocked.

### Stateful Packet Filtering Firewall

A stateful packet filtering firewall monitors the communication between the media player and the Internet to ensure that inbound packets are being sent at the request of a media player inside the firewall. Similar to packet filters, it may include additional options that allow more sophisticated actions to be taken with individual packets. These firewalls should be configured to permit RTSP and MMS traffic.

### Network Address Translation Firewall

A network address translation firewall converts the media player's internal address to an external address before it forwards the player's requests to Helix Server. Once it receives a request, Helix Server sends its UDP packets directly to the firewall, rather than to the player, and the firewall may not know which player requested the packets. Network address translation is often

implemented as part of packet filtering firewalls or stateful packet filtering firewalls.

### SOCKS Firewall

Only software with built-in SOCKS support, which must additionally be configured by the user, can send data through a SOCKS firewall. RealPlayer does not include SOCKS support.

**Tip:** In some cases, a user can install a Winsock.dll that supports SOCKS, and configure it to point to the SOCKS firewall.

### Summary of Firewall Configuration Requirements

The table below summarizes the six most common firewall types and any special configuration information.

**Streaming Media Over Different Types of Firewalls**

	Application- Level Proxy	Transparent Proxy	Packet Filter	Stateful Packet Filtering	Address Translation	SOCKS
Media player configuration required?	Yes	No	No	No	No	Yes
IP address seen by the media player.	Firewall	Server	Server	Server	Server	Firewall
IP address seen by the server.	Firewall	Firewall	Player	Player	Firewall	Firewall
Valid inside addresses required?	No*	No*	Yes	Yes	No*	No*
RTSP support required to get UDP?	Yes	Yes	No	No	Yes	No**
RTSP support required to get TCP?	Yes	No ***	No	No	No	No

\* Usually requires compliance with RFC 1597 “Address Allocation for Private Internets” (<http://www.ietf.org/rfc/rfc1597.txt>).

\*\* Requires SOCKS version 5.0.

\*\*\* May require special configuration.

### Addresses in Access Logs

Depending on the type of firewall and its location, the media player address shown in the access log may not reflect the true address. The following table

lists the address that will appear in the access log as the requesting media player's address.

#### Addresses Shown in Access Logs

	Application- Level Proxy	Transparent Proxy	Packet Filter	Stateful Packet Filtering	Address Translation	SOCKS
Address shown in Helix Proxy access log when a firewall is between Helix Proxy and the player.	Firewall	Firewall	Player	Player	Firewall	Firewall
Address shown in Helix Proxy access log when a firewall is between Helix Proxy and Helix Server.	Player	Player	Player	Player	Player	Player
Address shown in Helix Server access log when a firewall is between Helix Proxy and Helix Server.	Firewall	Firewall	Proxy	Proxy	Firewall	Firewall

## Default Ports

The following sections explain the default ports used by Helix Proxy when receiving requests and sending data. This information will help you to decide which ports to open on your firewall. Note that many port values, such as the RTSP and HTTP ports, are configurable.

**For More Information:** See also the RealNetworks firewall information at <http://service.real.com/firewall>. For more about changing default port values, refer to “Defining Communications Ports” on page 45.

## Media Players

This section covers ports used with media players, as well as with HTTP servers that host RDF files.

## Media Player Listen Ports

The following table lists the default ports that Helix Proxy uses to listen for media player requests.

**Default Listen Ports for Media Player Requests**

Activity	Port Number	Transport	Purpose
listen on	554	TCP	Control channel for RTSP requests. Data channel also, if TCP was requested.
listen on	80	TCP	HTTP requests, as well as RTSP and MMS cloaked through HTTP.
listen on	1755	TCP or UDP	TCP control channel for MMS requests. Data channel also, if TCP was requested. UDP resend requests by MMS.
listen on	6970-65535	UDP	Data channel for RealNetworks and RTP-based media players.
listen on	6970-65535	UDP	RDT/RTP client replies for UDP resends, etc.
listen on	1024-5000	UDP	Data channel for MMS media players.

## Media Player Data Ports

When Helix Proxy accepts a stream request, it directs outgoing data to a port specified by the media player. The following table lists the data port ranges used by media players. A firewall should not restrict outgoing data sent to these client ports.

**Default Data Ports on Media Players**

Activity	Port Number	Transport	Purpose
send to	6970-32000	UDP	Packet delivery for RealNetworks media players.
send to	1024-65535	UDP	Packet delivery for RTP-based media players.
send to	1024-5000	UDP	Packet delivery for MMS media players.

**Note:** If the client chooses TCP for the data channel, Helix Proxy uses the same port for both the control channel and the data channel.

**Tip:** In addition to these settings, RealPlayer inherits proxy settings (if any) from the default browser. Users can turn off this feature through the RealPlayer **Preferences** menu, however.

## HTTP Servers Hosting RDF Files

If you use the capabilities exchange feature for rate control, Helix Proxy needs to contact HTTP servers to obtain media player RDF files. To enable this, allow outgoing TCP communication to port 80 on external HTTP servers.

**For More Information:** Refer to “RDF Files” on page 65.

## Origin Servers

The following tables list the default ports used by Helix Proxy to communicate with an origin Helix Server.

**Default Ports Used with Helix Server**

Activity	Port Number	Transport	Purpose
send to	554	TCP	Control channel for RTSP requests to Helix Server version 9 and later.
send to	3030	TCP or UDP	Data and control channel for pull-splitting requests to RealSystem Server version 8 and earlier.
send to	7802, 7878	TCP	Cache requests to RealSystem Server version 8 and earlier.
listen on	6970-32000	UDP	Proxy data channel.

## Helix Administrator

The following table lists the default ports that Helix Proxy uses to receive requests from administrative tools.

**Default Ports for Administrative Tools**

Activity	Port Number	Transport	Purpose
listen on	admin port (random)	TCP	Helix Administrator requests.
listen on	9090	TCP	Proxy Monitor data.



## ACCESS CONTROL

Using the access control feature, you can limit access to Helix Proxy based on the IPv4 or IPv6 address of the requesting machine and the port to which the request is made. This chapter explains how to implement access control.

**Note:** To implement user name and password control for media media players, use the authentication feature, which is described in Chapter 11.

## Understanding Access Control

The access control feature associates permission to connect to certain ports with media player addresses. For example, you can allow only certain groups in your organization to view clips by giving those groups' IP addresses access to application protocol ports on Helix Proxy. If a media player requests a clip through a port for which it has no access, it receives a message that the URL is invalid, or that the connection has timed out.

### Rule Components

Helix Proxy uses rules to implement access control policy. Each access rule provides the following information:

- **Sorting Order**—Order in which a rule is implemented. Helix Proxy implements access rules in order, from the first to the last. This is important to keep in mind when establishing the order in which you wish your rules to apply.
- **Access**—Whether the media player is allowed or denied access.
- **Client IP Address**—Media Player's address, or a range of addresses. This can also be an encoder's IP address.
- **Server IP Address**—Helix Proxy's address.

- **Ports**—Port numbers on Helix Proxy to which access is allowed or denied. For general content viewing, these numbers correspond to the RTSP, HTTP, and MMS ports. For encoders, these correspond to the port numbers in the broadcasting setup pages.

## Predefined Access Rules

Helix Proxy predefines three access rules:

- **Allow localhost access**—This rule permits access to Helix Proxy from an application running on the same computer. You should not edit this rule. This rule should always come first in the access control list.
- **Deny connections to port 7070**—This rule prevents other computers from accessing port 7070, which is reserved for use by Helix Proxy. You should not edit this rule.
- **Allow all other connections**—This rule allows all media players to make any request on any port. Access is denied, though, if the content is secured, and the media player does not supply a valid user name and password.

By default, the third rule allows all media players to make requests on all ports. Hence, access control checking is off. To turn access control on, you need to delete or modify the third rule, and implement new rules.

## Access to Helix Administrator

When you implement access control, you may inadvertently lock yourself out of Helix Administrator by denying all access to the Admin port. Therefore, if you decide to set up access control, the first rule to define should allow access to the Admin Port. This rule needs to come directly after the predefined Allow localhost access rule. The section “Granting Access to Helix Administrator” on page 136 explains how to create this rule.

## Access Rule Methods

To use the access control feature, you must make decisions about the types of rules you will create. Then, you can create as many rules as you need. There are two general methods that you can use to restrict access to Helix Proxy:

- **specific address denial**

In this method, you deny access to a specific group of IP addresses and ports, and allow access to everyone else. This is the better policy if you

want to block a small number of clients, while allowing most clients to make requests.

- specific address permission

This method is the opposite of the preceding. Here, you allow access to a specific group of IP addresses and ports, and deny access to everyone else. This is the better policy if you want to block a large number of clients, allowing only a small number of clients to make requests.

When you create a rule, you select a specific client IP address. Optionally, you can extend the addressing by choosing a bit mask, as described in Appendix B. You then select the ports for which that set of clients is allowed or denied access. You may need only one access rule. Or, you may want to set up several.

## Rule Order

When you create multiple access rules, you set a rule order using the up and down arrow buttons on the rule list. Helix Proxy carries out rules in order from first to last. When a client connects, Helix Proxy evaluates the connection starting with the first rule on the list. As soon as it finds a rule that matches the player's address, it allows or denies access according to that rule.

**Tip:** When implementing an access control policy, make the rules at the top of the list more strict. Reserve lower positions for the more lenient rules.

## IPv4 and IPv6 Access Rules

Helix Server supports access rule checking for both IPv4 and IPv6 addresses. If Helix Server runs on a machine that has both IPv4 and IPv6 addressing, you may need to create rules for both IPv4 addresses and IPv6 addresses:

- Client connections using an IPv4 address are checked against IPv4-based rules.
- Client connections using an IPv6 address mapped to an IPv4 address are checked against IPv4-based rules.
- Client connections using an IPv6 address are checked against IPv6-based rules.

## Granting Access to Helix Administrator

If you decide to implement access control rules, the first step is to set up a rule that enables you to connect to Helix Administrator, regardless of the restrictions you create in other rules.

► **To grant access to Helix Administrator:**

1. If you do not know the Admin Port number, click **Proxy Setup>Ports**. Or, click the **View** link at the bottom of the Access Control page. Note the value of the **Admin Port** field.
2. Click **Security>Access Control**.
3. Click the “+” icon in the **Access Rules** section.
4. In the **Edit Rule Description** box, enter a rule description such as `AccessToAdmin`.
5. In the **Access Type** pull-down list, select **Allow**.
6. In the **Client Hostname or IP Address/Netmask** box, you have two choices:
  - a. Type **Any**. Although this appears to allow everyone access to Helix Administrator, administrator log-in is guarded by the randomly-generated Admin port number, as well as user name and password validation, as described in “Administrator Authentication” on page 140.
  - b. For additional security, specify the IP address for permitted users. You can indicate a range of allowable addresses by adding a net mask after the address, separating the two entries with a forward slash (/).

**For More Information:** For information on using a bit mask, see Appendix B.
7. In the **Server Hostname or IP Address** box, type **Any**.
8. In the **Ports** box, type the Admin Port number.
9. In the **Access Rules** area, click the up arrow to make the administrator access rule the third rule on the list.
10. Click **Apply**.
11. Restart Helix Proxy.

## Creating General Access Rules

Use the steps in this section to allow or deny access to specific IP addresses or address ranges.

**Warning!** Be sure to first follow the steps in “Granting Access to Helix Administrator” on page 136, or you will not be able to access Helix Administrator after you restart Helix Proxy.

► **To limit access according to IP number:**

1. Review the port numbers in use for RTSP (usually 554), and MMS (usually 1755). You can determine the port values by clicking **Proxy Setup>Ports**. Or, click the **View** link at the bottom of the Access Control page.
2. Click **Security>Access Control**.
3. Click the “+” icon and enter a short description for the new access rule in the **Edit Rule Description** box. This description is for your reference only.
4. From the **Access Type** list, indicate whether permission is being granted or refused by selecting Allow or Deny.
5. In the **Client Hostname or IP Address/Netmask** box, type the IPv4 or IPv6 address of the client machine. To indicate a range of client IP addresses, add a net mask after the address, separating the two entries with a forward slash (/). To refer to all clients regardless of IP address, enter Any.

**For More Information:** For information on using a net mask, see Appendix B. See “IPv4 and IPv6 Access Rules” on page 135 for information about rule-checking when an IPv6 address has a mapped IPv4 address.

6. In the **Server Hostname or IP Address** box, type the IPv4 address, IPv6 address, or host name of Helix Proxy. You can type a specific address, or use the word Any to refer to any IP address Helix Proxy uses to listen for incoming requests.

**Note:** If you type a specific IP address or host name rather than Any, ensure that the address is in the IP binding list. See “Binding To An IP Address” on page 48 for more information.

7. List the Helix Proxy port numbers to which you want to restrict access. In the **Ports** box, type the port numbers you noted in Step 1, separating entries with commas. For example, type the following:  
1090, 554
8. In the **Access Rules** area, click the up arrow or down arrow to move the rule to its appropriate position on the list. General access rules should always come after the Allow localhost access rule, and the rule you created for allowing access to Helix Administrator. For more information, see “Rule Order” on page 135.
9. Click **Apply**.
10. Restart Helix Proxy.

## AUTHENTICATION

Helix Proxy authentication provides a way to control what or who can access Helix Proxy, whether someone using Helix Administrator, or a user requesting streaming media. Through the authentication feature, you can configure Helix Proxy to require a valid user name and password before allowing a client to access a particular URL.

**For More Information:** To limit use of Helix Proxy based on bandwidth or connection volume, use the methods described in the section “Controlling Bandwidth” on page 52. Chapter 10 explains the access control feature, which lets you allow or deny access based on the media player’s IP address.

### Understanding Authentication

Authentication verifies the identity of the users or software programs that make requests of Helix Proxy. It usually takes the form of user name and password validation, though this is not necessary in all cases. The following sections describe the major authentication features and components.

#### Types of Authentication

There are several types access requests that you can authenticate, such as viewers requesting media, or Helix Proxy users logging into Helix Administrator.

##### Media Viewer Validation

Authentication verifies the identity of users who request content from Helix Proxy. The verification comes in the form of asking for a name and password. To receive requests on behalf of clients, Helix Proxy requires an accounting channel between the requesting client and itself. Helix Proxy uses the accounting channel to request and receive authentication information.

The following table lists supported media players and the types of authentication that you can use with them. Basic, RealSystem 5.0, and Windows NT LAN Manager are forms of user name and password validation, as described in “Authentication Protocols” on page 149.

**Media Players and Supported Authentication Types**

Media Player	Basic	RealSystem 5.0	Windows NT LAN Manager	player GUID
RealPlayer 3 and earlier	no	no	no	no
RealPlayer 4	no	no	no	<b>yes</b>
RealPlayer 5 and higher	<b>yes</b>	<b>yes</b>	<b>yes</b>	<b>yes</b>
Windows Media Player	no	no	no	no
QuickTime Player	<b>yes*</b>	no	no	no
Any other RTP-based player	no	no	no	no

\* – Basic authentication functions for QuickTime Player running on Windows only.

### Administrator Authentication

Accessing Helix Administrator requires a valid user name and password. As explained in “Starting Helix Administrator” on page 38, the URL used to access Helix Administrator contains the /admin/ mount point, which automatically authenticates the login. The installation process creates the initial user name and password pair, but you can add additional user names and passwords to the SecureAdmin realm, as described in “Managing Users and Passwords” on page 143.

### Media Requests Requiring Authentication

Helix Server secures content by placing clips in *protected paths*. Access to a protected path requires the use of a special mount point in the request URL, as shown in the following example:

```
http://helixserver.example.com/ramgen/secure/video1.rm
```

A viewer request containing the mount point to the protected path requires authentication. Helix Proxy uses its accounting connection to Helix Server to determine which paths are protected. It can then enforce authentication for any protected path on any Helix Server.

Typically, Helix Server also implements authentication. Users therefore may be asked for a user name and password from both Helix Proxy and Helix

Server. In each case, access to the protected content requires the viewer to supply the correct user name and password stored by each server. These user name and password combinations may be different.

## Authentication Components

As you set up authentication, you work with databases, which store privileges, as well as realms, which validate user names and passwords.

### Databases

On each authentication request, Helix Proxy verifies the user's password and permissions in a database. By default, Helix Proxy uses flat file databases, as described in "Using Databases" on page 147. It uses different databases for different types of authentication. One database holds permissions for media players, for example, while another verifies the identity of users accessing Helix Administrator.

To implement authentication on a limited scale (a few hundred users, for example), use the predefined flat file databases. This requires no additional database configuration. For large-scale implementations of authentication, however, you can tie Helix Proxy's authentication system to a relational database. On Windows NT/2000/XP systems, you can also tie authentication into an existing LAN manager database.

**For More Information:** For more about databases, see "Using Databases" on page 147. Appendix C explains the database structure, which you'll need to know to use a relational database for authentication. See also "Windows NT LAN Manager" on page 150.

### Realms

An authentication realm indicates the database that stores a user's name and password, and specifies the authentication protocol used to validate the user's identity. An authentication protocol, which is not related to streaming protocols such as RTSP, determines how passwords are encrypted in the database. You can use a basic encryption protocol, or a more secure protocol that works with RealNetworks media players only.

**Tip:** Depending on your authentication needs, you may not need to change or add to the predefined realms. For more on realms, see "Setting Up Realms" on page 149.

## Setting up Authentication

To set up authentication in Helix Proxy, you need to turn on the feature, and decide which realm and database to use with authentication. Optionally, you can select sites all users are allowed to visit, and allow users to view content from more than one location.

► **To enable authentication:**

1. In Helix Administrator, click **Security>Authentication**.
2. From the **Enable Authentication** list, select Yes.
3. From the **Realm** pull-down list, select ConnectRealm. If you have set up another realm, select that name here. For more on realms, refer to “Setting Up Realms” on page 149.
4. From the **Database** pull-down list, select Connect\_RN5. If you have set up another database, select that name here. For more on databases, see “Using Databases” on page 147.

**Note:** If the selected realm uses Windows NTLM as an authentication protocol, select **None** for the database.

5. To enable each user to view content from more than one media player, set **Allow Duplicate IDs** to Yes. If **Allow Duplicate IDs** is set to No, a user can view a given clip from only one computer at a time. The user receives an error message if he or she tries to view the content from a second computer without logging out at the first location.

**Tip:** You can use the duplicate ID option to grant access to groups. For example, you could set **Allow Duplicate IDs** to Yes and assign all employees in a certain department the same user name and password. The entire department can then use this single account to view content.

6. Optionally, you can choose sites from which users can request content without having to supply a user name and password. Follow these steps to set up non-authenticated sites:
  - a. In the **No-Authenticate Rules** area, click the “+” icon. A generic rule name appears.
  - b. In the **Edit Rule Name** box, type a name for this rule.

- c. In the **Host** box, type the name of the site to which all users will be permitted access. Use a single asterisk as a wildcard, such as in the following examples:

*.org	All sites ending with .org.
example.com	The site named www.example.com.
*.example.com	Any site ending in example.com, such www.example.com and sports.example.com.

**Note:** Only one asterisk is allowed per rule. For example, \*.\*.com is not permitted.

7. Click **Apply**.

## Managing Users and Passwords

The following sections explain how to manage the user names and passwords, whether for individuals requesting secured clips, or for users of Helix Administrator. Note the following points first, however, before defining user names and passwords:

- To validate access attempts from QuickTime Player, set the Basic authentication protocol in your authentication realms. For instructions on doing this, see “Creating or Modifying a Realm” on page 151.
- If you are using Windows NT/2000/XP to manage the list of users, passwords, and groups, use those tools instead of the instructions below. To use Windows passwords, you need to set the NTLM authentication protocol in your selected realm, as described in “Creating or Modifying a Realm” on page 151.

### Adding a User

Follow the procedure below to add a user and password to an authentication realm. To use a database other than a predefined, flat file, you must create that database and associate it with the proper realm before adding users. Refer to “Using Databases” on page 147 and “Setting Up Realms” on page 149 for more information.

**Note:** The Helix Administrator interface does not provide a way to add multiple user names and passwords at one time.

► To add a user name and password:

1. Click **Security>Realms**.
2. In the **Authentication Realms** list, select the realm to which you want to add a user. The following realms are predefined:

ConnectRealm	media player users
SecureAdmin	Helix Administrator users

**For More Information:** Realms are described in “Setting Up Realms” on page 149.

3. Click **Add a User to Realm**.
4. In the pop-up window, define the user’s name in the **Name** box. User names are case-sensitive. You can use separate words if, for example, you want to use full names of users.
5. In the **Password** box, supply the user’s password. Passwords are case-sensitive. RealNetworks recommends following good password practices:

- Avoid common words that are easy to guess.
- Do not use a word associated with the user, such as a first name.
- Do not use the same password for multiple users.
- For highest security, use a random combination of letters and numbers in different cases.

**Tip:** Keep track of the passwords you assign. Helix Administrator allows you to change passwords, but not to look them up.

6. In the **Confirm Password** box, type the password again.
7. Click **OK**.

## Deleting a User

The following procedure explains how to delete a user from a database. Helix Administrator does not have a bulk delete feature.

► To remove a user:

1. Click **Security>Realms**.

2. In the **Authentication Realms** list, select the name of the realm in which you want to delete a user. The following are the default realms:
 

ConnectRealm	media player users
SecureAdmin	Helix Administrator users
3. Click **Remove a User from Realm**.
4. In the new window that appears, enter the user's name in the **Name** box.
5. Click **OK**.

### Browsing All User Names

The browsing feature lists all user names defined for an authentication realm.

► To browse all users:

1. Click **Security>Realms**.
2. In the **Authentication Realms** list, select the realm you want to browse. The following are the default realms:
 

ConnectRealm	media player users
SecureAdmin	Helix Administrator users
3. Click **Browse Users in Realm**. The pop-up window lists all user names defined for that realm.

### Changing a Password

The following procedure explains how to change the password for an existing user. The Helix Administrator interface does not allow you to look up existing passwords.

► To change a password:

1. Click **Security>Realms**.
2. In the **Authentication Realms** list, select the name of the realm that contains the user. The following are the default realms:
 

ConnectRealm	media player users
SecureAdmin	Helix Administrator users
3. Click **Change User Password**.

4. In the new window that appears, enter the user's name in the **Name** box.
5. In the **Password** box, specify the user's new password.
6. In the **Confirm Password** box, type the password again.
7. Click **OK**.

## Using the Password Tool

When it uses the RealSystem 5.0 authentication protocol, Helix Proxy encrypts passwords, so you cannot look up the passwords directly. However, you can add or change passwords in a flat file or relational database by using a command-line utility. You can even create a password interface by integrating this utility with your own CGI scripts and Web pages.

**For More Information:** See "Authentication Protocols" on page 149.

### ► To use the password tool:

1. Open a command prompt and navigate to the Bin directory under Helix Proxy's main installation directory.

2. Enter the following command:

```
makepass username realm
```

using the following values:

*username* The user name exactly as it is entered, or will be entered, in the authentication database.

*realm* The value of the **Realm** variable specified in the relevant list. For Helix Administrator users, use the value of the Realm variable in the RealAdministrator\_Files list within the FSMount list in the configuration file.

3. A password prompt appears, followed by a prompt to type the password again. The resulting encrypted password is displayed on the screen.

Helix Proxy encrypts passwords with the MD5 hashing algorithm. It uses the form MD5("username:realm:new\_password"). On BSD systems and some other UNIX systems, you can generate these passwords with the following command:

```
echo -n "username:realm:new_password" | md5
```

4. Add the resulting encrypted password into the appropriate field of the database:
  - For text files, place it in the password field of the Users directory. See “Users Directory” on page 230.
  - For databases, place it in the password field of the Users table. See “Users Table” on page 232.

## Using Databases

In its default configuration, Helix Proxy uses flat files to store user names, passwords, and permissions. For large-scale implementations of authentication, RealNetworks recommends that you tie Helix Proxy into a relational database that stores this information. The following table lists the flat file databases automatically installed with Helix Proxy.

**Default Databases**

Name	User Names and Passwords	Purpose
Admin_Basic	Helix Administrator.	Validate access to Helix Administrator.
Connect_RN5	content users	Validate users requesting secured on-demand or live content.

## Supported Database Types

Helix Proxy provides interfaces to several types of databases. Appendix C contains details about the database structure, which you’ll need to know to integrate Helix Proxy’s authentication system with a relational database, for example.

### Flat File Database

The default databases used for authentication are flat text files, which work well for storing relatively small amounts of data, such as a few hundred user names and passwords. You may want to use them to learn the authentication data structure before linking Helix Proxy to a more robust relational database. If you choose to use the default flat files exclusively, you do not need to perform any additional configuration.

## ODBC

Helix Proxy includes templates for ODBC-compliant databases. To use an ODBC database, you must configure your database to comply with the appropriate table structure described in Appendix C.

## mSQL

Helix Server includes templates for mSQL databases. To use an mSQL database, you must configure your database to comply with the appropriate table structure described in Appendix D.

## Adding a Database

Follow the procedure below to add a new database that stores user names and passwords. If you are using the default flat file databases, it is typically not necessary to add a new database.

► To add a database:

1. Click **Security>User Databases**.
2. Click the “+” icon, and type a description for the new database in the **Edit Database Name** box.
3. From the **Database Type** list, select the data storage method you want to use. Database types are described in “Supported Database Types” on page 147.
4. Depending on the database type method you chose, additional information is required.
  - a. **Flat File** needs only the path to the main text file directory. For example, the `con_r_db` directory under the main Helix Server directory. For more information, see “Understanding Authentication Data” on page 229.
  - b. **ODBC** databases use the following items:

Database Name	Name of the database.
Table Name Prefix	Prefix used to make field names unique, when used with an existing database.
User Name	Name required by the database application.
Password	Password required by the database application.
Confirm Password	

**For More Information:** Refer to “Setting Up Other Types of Data Storage” on page 233 for further instructions.

c. **mSQL** databases require the following items:

Host Name	IP address or DNS name of computer where the database is stored.
Database Name	Name of the database.
Table Name Prefix	Prefix used to make field names unique, when used with an existing database.
User Name	Name required by the database application.
Password	Password required by the database application.
Confirm Password	

5. Click **Apply**.

## Setting Up Realms

A realm connects users to databases. When you define passwords, you add them to a realm. The realm, in turn, specifies the encryption protocol, and indicates the database where information is stored. If you are using the default authentication databases, as described in “Using Databases” on page 147, you can use the default realms, too, changing the authentication protocols if necessary. If you set up new databases, you need to create new realms, or point the existing realms to your new databases.

**Predefined Authentication Realms**

Realm	Authenticates	Realm ID	Protocol	Database
ConnectRealm	content users	<i>servername.</i> ContentRealm	RealSystem 5.0	Connect_RN5
SecureAdmin	Helix Administrator	<i>servername.</i> AdminRealm	Basic	Admin_Basic

## Authentication Protocols

Authentication protocols determine the password encryption and storage method used by Helix Proxy. The server supports three protocols. Each realm uses just one protocol.

## Basic

The Basic protocol sends the user's name and password over the public Internet in a simple manner, encoding them with the Base64 algorithm. Helix Proxy decodes and verifies the password. Information can be stored in a flat file or a relational database. This protocol works with RealNetworks media players, as well as the QuickTime Player.

## RealSystem 5.0

Also called *RNS*, this is a RealNetworks encryption protocol that works with RealPlayer 5 and later. It is more sophisticated and secure than the Basic protocol. Use it if your material will be served exclusively to users who have RealPlayer 5 or later. Information can be stored in a flat file or a relational database.

**Tip:** For authentication of QuickTime Player or RealNetworks players earlier than version 5, use the Basic protocol.

## Windows NT LAN Manager

The NTLM method is suitable for a Windows-based intranet. It enables Helix Proxy to use the existing Windows NT database of user groups. It also allows access control of content through NTFS file permissions. This method requires that Helix Proxy itself be installed on the Windows NT/2000/XP machine. When using NTLM authentication, be aware of the following:

- NTLM authentication works only with RealPlayer 5 and later.
- All users' accounts must exist on the local computer. NTLM authentication will not work with accounts on other servers within the domain, but it will authenticate against accounts on the primary domain controller.
- You add all user accounts through the Windows NT User Manager. Do not use the instructions in "Managing Users and Passwords" on page 143.
- The built-in guest account is not available for use in authentication.
- When you select NTLM authentication for Helix Administrator access, all groups are authenticated if you do not specify a user group.
- You cannot evaluate permissions on commerce rules when you use NTLM authentication.
- Blank passwords are not supported.

## Creating or Modifying a Realm

The following procedure explains how to create a new realm or modify an existing one. It is generally not necessary to do this unless you have created a new database. Use a one-to-one correspondence between realms and databases. Do not create two realms, for instance, that use the same database.

► **To create or modify a realm:**

1. Click **Security>Realms**.
2. To create a realm, click the “+” icon and enter a name for this realm in the **Edit Realm Description** box. To modify an existing realm, select it in the **Authentication Realms** box.
3. In the **Realm ID** box, type a name that will be used in other areas of Helix Administrator. The realm name may also appear to users as part of the name and password prompt. The default realms conform to the following format:

*servername.Realm\_Id*

**Warning!** You do not have to use the default convention, but you must include a period (.) in the realm ID or the realm will not work properly.

4. In the **Authentication Protocol** list, select the authentication method you want to use for this realm, as described in “Authentication Protocols” on page 149.
  - a. If you choose Basic or RealSystem 5.0, select the database in the **Database** box.
  - b. If you choose Windows NT LAN Manager, Helix Server uses the NT list of names instead of a database. Type the appropriate provider in the **Provider** list, such as NTLM. Type the Group name in the **Group** box.
5. Click **Apply**.



## MONITORING

This section deals with compiling statistics, creating reports, monitoring network connections, and troubleshooting Helix Proxy. You can create real-time reports about media streamed by Helix Proxy, for example, or just compile error messages in a simple log file.



## BASIC LOGGING

This chapter explains how Helix Proxy records information about client connections and other events. Using the log files, you can compile reports about system activity, gathering the statistical information you need.

**Tip:** If you're interested in designing custom reports to track specific activities on Helix Proxy, refer to Chapter 13.

### Understanding Basic Logging

Helix Proxy maintains a basic access log that includes statistics about client connections. It keeps another log of error and informational messages about Helix Proxy operation. The log files are text files that you can open with any text editor, or parse with a script or application. As accesses or errors occur, Helix Proxy appends information to the end of the log file. The following sections introduce you to the log files and their features.

#### Basic Access Log

The basic access log records information about requests by RealNetworks media players, Windows Media Player, and QuickTime Player. Using these logs, you can find out what clips were played, the times when media players connected, and so on. This information can help you determine which clips are most popular, for example. The default access log is `proxy.log`, which is located in the `Logs` subdirectory of the main Helix Proxy installation directory.

#### Logged Information

Helix Proxy provides eight logging styles that determine the amount of information gathered on each access attempt. In general, each style builds on the preceding style, adding more information. For instance, logging style 0 gathers the least amount of information. Logging style 1 includes the style 0

information, and adds more information, and so forth. You choose just one logging style for the entire log.

**For More Information:** The section “Basic Access Log File Format” on page 158 explains the logging styles and information fields.

### Media Player Statistics

All logging styles can record statistics about a media player’s playback experience. These statistics let you learn how many media packets were dropped, for instance, or whether the viewer paused the clip. There are four types of client statistics. You can use any combination of these statistics types, up to all four. Or, you can turn off client statistics gathering entirely. As well, users may choose not to report statistics.

**Note:** Only RealNetworks desktop media players report all of the client statistics. Other desktop media players may report some of the statistics.

**For More Information:** See “Client Statistics” on page 171.

### Server and Proxy Access Logs

Helix Server records similar information about client requests in its basic access log. Additionally, it records proxy information, such as the proxy IP address, when a clip is delivered by either proxy pull-splitting or proxy cache. Helix Server and Helix Proxy access log settings are independent. For example Helix Proxy can record information with logging style 0 whereas Helix Server can collect its information using logging style 5.

### Basic Error Log

The basic error log contains information and error messages about Helix Proxy operation. By looking for patterns of errors, you can troubleshoot and correct possible problems on your site. The default file name is proxyerr.log, and the file is generated in the Logs subdirectory of the main Helix Proxy installation directory. Helix Proxy records an entry in this log only when an error occurs. Until an error happens, the file does not exist. The error log uses the following syntax:

```
***date time proxyname(process_ID): error_message
```

The following table explains these fields in the error log file.

<b>Error Log Fields</b>	
Entry	Meaning
***	Three asterisks indicate an error. Informational messages are not preceded by asterisks.
date	Date on which the error occurred, given in the form dd-Mmm-YY, as in 26-Apr-02.
time	Time the error occurred on the Helix Proxy clock, given in the form HH:MM:SS.xyz, as in 21:05:10.614.
proxyname(process_ID)	Helix Proxy name, followed by the process ID in parentheses.
error_message	Text of the message.

**Note:** If you receive a message that refers to a fatal error, contact RealNetworks Technical Support for assistance.

**For More Information:** For details about error messages recorded in the error log, refer to *Helix Server and Helix Proxy Troubleshooting Guide*.

## Log File Rolling

The access and error log files can grow indefinitely as they accumulate data. To keep log files manageable, you can limit a log file to a specific size. With the access log, you can also create a new log at a preset interval, such as every six hours or two weeks, depending on the amount of data you expect to log. Helix Proxy begins, or *rolls*, a new log file when the limit is reached. Rolled log files are named with the following format:

*file\_name.log.timestamp*

The name and extension are set through Helix Administrator, as described in “Customizing the Access and Error Logs” on page 179. The timestamp has the following format, using a 24-hour clock:

YYYYMMDDHHMMSS

For example, the following file was created on June 22, 2008, at 1:49.53 P.M:  
proxy.log.20080622134953

## Basic Access Log File Format

Helix Proxy records each access request in a separate record written to a new line in the access log. Fields within a record are separated by spaces or by pipes (|). One record is created for every clip served. If the client requests a presentation that includes several clips, one record is created for each clip in the presentation.

### Logging Style

Helix Proxy provides eight logging styles, numbered 0 through 7. Styles 1 through 4 each include the information of lower logging styles. For example, style 3 collects the same information as styles 0, 1, and 2, as well as some additional information. The default is style 5, which adds a `presentation_ID` field to the information in style 2. The following sections describe which fields each logging style collects.

**Tip:** Although square brackets in syntax examples typically indicate optional material, the square brackets shown in the following access log syntax actually appear in the access log records.

**Note:** In the following examples, client statistics are not logged, so each entry shows `[UNKNOWN]` where the statistics fields would be. If you collect client statistics, therefore, each log entry will contain additional information. For more information, see “Client Statistics” on page 171.

**For More Information:** The section “Access Log Fields” on page 161 explains the information logged in each field.

#### Logging Style 0

Logging style 0 uses this format:

```
client_address - - [timestamp] "GET filename protocol/version" HTTP_status_code
bytes_sent [client_info] [client_stats_results] [proxy_info]
```

Here is an example of an actual log record, showing that 858,636 bytes of the requested clip were sent over RTSP:

```
207.188.7.125 - - [26/Jun/2008:10:31:44 -0700] "GET real9video.rm RTSP/1.0"
200 858636 [WinNT_5.0_6.0.10.714_RealPlayer_RN92PD_en_686] [UNKNOWN]
[Demand Cache Hit]
```

### Logging Style 1

Logging style 1 uses the following format:

```
client_address - - [timestamp] "GET filename protocol/version" HTTP_status_code
bytes_sent [client_info] [client_stats_results] file_size file_time connected_time
resends failed_resends [proxy_info]
```

The following sample log record shows the same information as logging style 0, but adds information on file size, clip timeline length, actual time streamed, and resent packages:

```
207.188.7.125 - - [26/Jun/2008:10:06:33 -0700] "GET real9video.rm RTSP/1.0"
200 858636 [WinNT_5.0_6.0.10.714_RealPlayer_RN92PD_en_686] [UNKNOWN]
926322 217205 1 0 [Demand Cache Hit]
```

### Logging Style 2

This is the format for logging style 2, which is identical to style 1, except that it records a client ID, which may be a global ID or an ID set through a cookie:

```
client_address - - [timestamp] "GET filename protocol/version" HTTP_status_code
bytes_sent [client_info] [client_ID] [client_stats_results] file_size file_time
connected_time resends failed_resends [proxy_info]
```

Here is an example:

```
207.188.7.125 - - [26/Jun/2008:10:07:42 -0700] "GET real9video.rm RTSP/1.0"
200 858636 [WinNT_5.0_6.0.10.714_RealPlayer_RN92PD_en_686]
[8e07b707-19b7-448b-96b6-96c90151f2a6] [UNKNOWN] 926322 217 205 1 0
[Demand Cache Hit]
```

### Logging Style 3

Logging style 3 follows this format. It builds on style 2 by adding information about the streams and the Helix Server or parent Helix Proxy that delivered the clip:

```
client_address - - [timestamp] "GET filename protocol/version" HTTP_status_code
bytes_sent [client_info] [client_ID] [client_stats_results] file_size file_time
connected_time resends failed_resends [stream_components] [start_time]
server_address [proxy_info]
```

This example shows the origin server and stream information added to the end of the record:

```
207.188.7.125 - - [26/Jun/2008:10:09:09 -0700] "GET real9video.rm RTSP/1.0"
200 858636 [WinNT_5.0_6.0.10.714_RealPlayer_RN92PD_en_686]
[8e07b707-19b7-448b-96b6-96c90151f2a6] [UNKNOWN] 926322 217 205 1 0
[1 1 0 0] [26/Jun/2008:10:05:14] 208.147.89.157 [Demand Cache Hit]
```

## Logging Style 4

Logging style 4 adds information about the clip's average bit rate and number of packets sent:

```
client_address - - [timestamp] "GET filename protocol/version" HTTP_status_code
bytes_sent [client_info] [client_ID] [client_stats_results] file_size file_time
connected_time resends failed_resends [stream_components] [start_time]
server_address average_bitrate packets_sent [proxy_info]
```

Here is an example:

```
207.188.7.125 - - [26/Jun/2008:10:10:04 -0700] "GET real9video.rm RTSP/1.0"
200 858636 [WinNT_5.0_6.0.10.714_RealPlayer_RN92PD_en_686]
[8e07b707-19b7-448b-96b6-96c90151f2a6] [UNKNOWN] 926322 217 205 1 0
[1 1 0 0] [26/Jun/2008:10:05:14] 208.147.89.157 34816 488 [Demand Cache Hit]
```

## Logging Style 5

Logging style 5, which is the default style, does not build on the preceding styles. Instead, it copies style 2 and adds a presentation ID that helps you keep track of presentations that contain multiple clips:

```
client_address - - [timestamp] "GET filename protocol/version" HTTP_status_code
bytes_sent [client_info] [client_ID] [client_stats_results] file_size file_time
connected_time resends failed_resends presentation_ID [proxy_info]
```

The following is an example of a logging style 5 entry:

```
207.188.7.125 - - [26/Jun/2008:10:11:03 -0700] "GET real9video.rm RTSP/1.0"
200 858636 [WinNT_5.0_6.0.10.714_RealPlayer_RN92PD_en_686]
[8e07b707-19b7-448b-96b6-96c90151f2a6] [UNKNOWN] 926322 217 205 1 0 124
[Demand Cache Hit]
```

## Logging Style 6

Logging style 6 includes all of the fields found in logging style 4. To these, it adds the presentation ID found in logging style 5, and appends two additional fields to the end of the entry:

```
client_address - - [timestamp] "GET filename protocol/version" HTTP_status_code
bytes_sent [client_info] [client_ID] [client_stats_results] file_size file_time
connected_time resends failed_resends [stream_components] [start_time]
server_address average_bitrate packets_sent presentation_ID
bitrate_adaptations media_adaptations [proxy_info]
```

Here is an example:

```
207.188.7.125 - - [26/Jun/2008:10:10:04 -0700] "GET real9video.rm RTSP/1.0"
200 858636 [WinNT_5.0_6.0.10.714_RealPlayer_RN92PD_en_686]
[8e07b707-19b7-448b-96b6-96c90151f2a6] [UNKNOWN] 926322 217 205 1 0
[1 1 0 0] [26/Jun/2008:10:05:14] 208.147.89.157 34816 488 124 1 0 [Demand
Cache Hit]
```

### Logging Style 7

Logging style 7 includes all of the fields of logging style 6. To the end of the log entry, it appends three fields used with fast channel switching.

```
IP_address - - [timestamp] "GET filename protocol/version" HTTP_status_code
bytes_sent [client_info] [client_ID] [client_stats_results] file_size file_time
connected_time resends failed_resends [stream_components] [start_time]
server_address average_bitrate packets_sent presentation_ID
bitrate_adaptations media_adaptations session_control_ID switch_count
final_switch [proxy_info]
```

Here is an example that includes the fields for fast channel switching:

```
207.188.7.125 - - [26/Jun/2008:10:10:04 -0700] "GET realvideo10.rm RTSP/1.0"
200 858636 [WinNT_5.0_6.0.10.714_RealPlayer_RN92PD_en_686]
[8e07b707-19b7-448b-96b6-96c90151f2a6] [UNKNOWN] 926322 217 205 1 0
[1 1 0 0] [26/Jun/2008:10:05:14] 208.147.89.157 34816 488 124 1 0 a345GHI987
5 1 [Demand Cache Hit]
```

**Note:** Because fast channel switching information is recorded only on the origin server, the channel switching fields are always blank in the proxy log. This makes logging style 7 essentially identical to logging style 6.

## Access Log Fields

The following table summarizes the various logging fields that may appear in an access record, and indicates which logging styles include the fields. The following sections describe the access log fields in detail.

**Access Log Fields**

Log Field	Logging Styles	Reference
client_address	0, 1, 2, 3, 4, 5, 6, 7	page 162
[timestamp]	0, 1, 2, 3, 4, 5, 6, 7	page 162
"GET filename protocol/version"	0, 1, 2, 3, 4, 5, 6, 7	page 163
HTTP_status_code	0, 1, 2, 3, 4, 5, 6, 7	page 163

(Table Page 1 of 2)

**Access Log Fields (continued)**

Log Field	Logging Styles	Reference
bytes_sent	0, 1, 2, 3, 4, 5, 6, 7	page 163
[client_info]	0, 1, 2, 3, 4, 5, 6v	page 163
[client_ID]	2, 3, 4, 5, 6, 7	page 164
[client_stats_results]	1, 2, 3, 4, 5, 6, 7	page 165
file_size	1, 2, 3, 4, 5, 6, 7	page 166
file_time	1, 2, 3, 4, 5, 6, 7	page 166
connected_time	1, 2, 3, 4, 5, 6, 7	page 166
resends	1, 2, 3, 4, 5, 6, 7	page 166
failed_resends	1, 2, 3, 4, 5, 6, 7	page 166
[stream_components]	3, 4, 6, 7	page 166
[start_time]	3, 4, 6, 7	page 167
server_address	3, 4, 6, 7	page 167
average_bitrate	4, 6, 7	page 167
packets_sent	4, 6, 7	page 167
presentation_ID	5, 6, 7	page 167
bitrate_adaptations	6, 7	page 167
media_adaptations	6, 7	page 168
session_control_ID	7	page 168
switch_count	7	page 168
final_switch	7	page 168
proxy_info	1, 2, 3, 4, 5, 6, 7	page 168

(Table Page 2 of 2)

**Client Address**

The `client_address` field gives the IP address of the client, such as 123.45.123.45. Following the IP address are two hyphens for compatibility with standard Web server log formats.

**Timestamp**

The `[timestamp]` field indicates the time that the record was written to the log file according to the Helix Proxy clock. It uses the following format:

`[dd/Mmm/yyyy:hh:mm:ss TZ]`

Here, TZ is the time zone expressed as the number of hours relative to Coordinated Universal Time (Greenwich, England). For example:

```
[26/Jun/2003:10:10:04 -0700]
```

### File Name and Protocol

The "GET filename protocol/version" field lists the file name and path requested by the client. The path is everything in the URL after the port number. If the client requests a file that doesn't exist, UNKNOWN appears in place of the file name. Possible values for the application-layer protocol used to send the clip to the client are RTSP and MMS. In addition, a letter at the end of the string indicates which transport type was used:

(blank)	UDP connection
T	TCP connection
M	Multicast

For example, RTSPT means that the clip was streamed using the RTSP protocol over a TCP connection. The version number indicates the edition of the protocol.

**For More Information:** See "GET Statements" on page 169.

### HTTP Status Code

The HTTP\_status\_code field holds a return code that uses the HTTP standard error codes. It usually returns 200.

### Bytes Sent

The bytes\_sent field records the number of bytes transferred to the client by any type of proxy delivery: pass-through, pull-split or cache mode.

### Client Information

The [client\_info] field describes the version and type of client being used.

#### RealNetworks Clients

For RealNetworks clients, [client\_info] uses the following format:

```
[platform_version_client_type_distribution_language_CPU]
```

The following information is recorded:

<i>platform</i>	Operating system client software runs on, such as WinNT, Mac, and so on.
<i>version</i>	Operating system version number.
<i>client</i>	Version number of the client software.
<i>type</i>	Type of client software.
<i>distribution</i>	Distribution code of the client software.
<i>language</i>	Language setting in client software.
<i>CPU</i>	Type of processor on which the client is running. If the processor does not have a hardware Floating Point Unit, the string no-FPU is appended to the end of the CPU field with no delimiter.

For example:

```
[WinNT_5.0_6.0.10.714_RealPlayer_RN92PD_en_686]
```

**Note:** RealAudio Player 1 logs just two fields for [client\_info]. They are *platform* and *client*.

#### Windows Media Player

For Windows Media Player, the [client\_info] field records the player version like this:

```
[NSPlayer/7.1.0.3055]
```

#### QuickTime Player

For QuickTime Player, the client information records the player version and the operating system. For example:

```
[QTS (qtver=5.0.2;os=Windows NT 5.0)]
```

#### Unknown Clients

If client information can't be gathered because the request came from a client that chose not to send statistics, [UNKNOWN] appears in the [client\_info] field.

### Client Identifier

For [client\_ID], the access log can record an identification number for each media player. This is a globally unique ID. (Because Helix Proxy typically resides behind a firewall, it does not attempt to gather cookie-based ids for clients.) The following sections explain the possible field entries.

### Globally Unique Identifier (GUID) for RealNetworks Client

The default settings for Helix Proxy and RealNetworks clients record a global ID for each client access attempt. Users can control whether GUIDs are transmitted, however. As well, you can disable the logging of GUIDs through Helix Proxy regardless of client configurations.

**For More Information:** See “Modifying the Basic Access Log” on page 179 for instructions on turning off client ID logging.

If a RealNetworks client is configured to send a globally unique ID, it does so. For privacy protection, however, RealPlayer is set by default *not* to send a GUID. Because sending a GUID rests solely at the discretion of each user, users must change their default GUID settings for their GUIDs to appear in the access logs. In RealPlayer, the user command for controlling GUID reporting is **Tools>Preferences>Connection>Internet Settings**.

**For More Information:** To review RealNetworks’ Consumer Software Privacy Statement, see the Web page located at **<http://www.realnetworks.com/company/privacy/software.html>**

### Windows Media Player and QuickTime Player IDs

If Windows Media Player and QuickTime Player are configured to send their GUIDs, Helix Proxy records those ID values. If the players do not send GUIDs, Helix Proxy generates an ID for the log. In this case, the same media player may be identified by multiple IDs in the log.

### Unknown IDs

When Helix Proxy can’t gather an ID because the client does not support GUIDs, empty square brackets—[ ]—appear in the [client\_ID] field. If GUID reporting is disabled on the proxy or media player side, the [client\_ID] field shows a series of zeroes instead of a unique client identifier:

```
00000000-0000-0000-0000-000000000000
```

### Statistics Results

The [client\_stats\_results] field holds connection statistics sent by the client when it finishes playing a clip, as described in “Client Statistics” on page 171. If the client blocks connection statistics, or the statistics cannot be collected, the field appears as [UNKNOWN].

## File Information

The `file_size`, `file_time`, and `connected_time` fields hold information about the requested clip or broadcast.

### `file_size`

The `file_size` field lists the size of the file as reported by the Helix Proxy operating system. This reported size includes the media data as well as the file header and other non-media information. For live broadcasts, `file_size` is always 0.

### `file_time`

The `file_time` field gives the total length, in seconds, of media stored in the media file. For live broadcasts, `file_time` is always 0. For SMIL files, this is always 20.

### `connected_time`

The `connected_time` field (formerly called “`sent_time`”) expresses in seconds how long the media player was connected to the server. Because RealNetworks media players close the connection when the clip reaches the end of its timeline or the viewer stops the clip, the `connected_time` value expresses the duration of the streaming session. Other media players, including QuickTime Player and Windows Media Player, keep the connection open until the viewer chooses another clip or closes the player. The values recorded for these players may therefore include time after which the clip stopped but the player remained idle.

## Resend Information

The `resends` field lists the number of packets successfully resent because of transmission errors. The `failed_resends` field gives the number of packets not successfully resent in time to correct transmission errors.

## Stream Components

The field `[stream_components]` is recorded only for RealNetworks media players. It explains the type of material sent, indicated in the following pattern:

`RealAudio_stream RealVideo_stream Event_stream Image_maps`

A value of 1 indicates that the clip includes this type of stream. The value 0 indicates that it does not. Thus, a clip that includes RealVideo and RealAudio but no event streams or image maps would appear in the access log as this:

1 1 0 0

### Start Time

The [start\_time] field gives the timestamp of when the clip began to stream, according to the Helix Proxy clock. It is identical in format to the timestamp at the beginning of each access record, but does not list the time difference from Coordinated Universal Time. Here is an example:

[26/Jun/2008:10:05:14]

### Server Address

The server\_address field lists the IP address of the Helix Server or Helix Proxy that delivered the clip. This may be the origin Helix Server, a Helix Server which is acting as a receiver, or another Helix Proxy which is acting as a proxy receiver.

In proxy cache mode, RTSP requests will show the cache's address (usually 127.0.0.1). To find the address of the origin Helix Server, look in the GET field (see "GET Statements" on page 169).

### Average Bit Rate

The average\_bitrate field lists the average bit rate of the clip in bits per second.

### Packets Sent

The packets\_sent field lists the total number of packets sent to the client.

### Presentation ID

The presentation\_ID field records a number used by all clips in the same SMIL or Ram presentation. SMIL files are also included in the log, and use the same number as their clips. For example, if the log entries for a SMIL file, a video clip, and a GIF image all list presentation ID 437, you can conclude that the SMIL presentation consisted of that video and image. Helix Proxy assigns the IDs, which are recorded only with logging styles 5 and 6, when it transmits the clips.

### Bit Rate Adaptations

The bitrate\_adaptations field records an integer value that indicates how many times the stream speed upshifted or downshifted during the playback session. This occurs only in clips that encode multiple bit rates, such as SureStream RealVideo. For example, if a media player connection first uses a 350 Kbps

stream, drops to a 225 Kbps stream, then returns to the 350 Kbps stream, the field value is 2, indicating two shifts in encoding speed. A value of 0 means that no bandwidth shifting occurred, or that the media clip is not encoded for multiple bit rates.

**Tip:** These values indicate the basic quality of service. Recurring, high numbers may indicate persistent network congestion problems.

### Media Format Adaptations

The `media_adaptations` field is used with media formats that support multiple codecs for the same clip. The field value is an integer that indicates the number of times that the media player shifted between the different formats. If a presentation starts out streaming a high-speed MPEG-4 encoding, for example, before shifting to a lower-speed H.263 encoding, the field value is 1. A value of 0 means that no media format shifting occurred, or that the media clip was encoded using a single codec.

**Note:** Shifting between media formats in a clip is not currently supported by Helix Proxy.

### Session Fields

The session fields provide information about RTSP sessions that delivered multiple streams. These fields are primarily of use when using channel switching. Because Helix Proxy does not record information about individual streams within the RTSP session, these fields do not contain meaningful information:

- `session_control_ID`
- `switch_count`
- `final_switch`

### Proxy Information

The `proxy_info` field gives information about the type of proxied stream, whether live or on-demand, and tells how Helix Proxy delivered the media

stream, such as by pass-through, pull-split or cache mode. One of the following values is recorded:

Accounting Only	Only accounting data (no media) was sent.
Demand Cache Hit	The proxied stream was an on-demand clip, and Helix Proxy served it from the media cache.
Demand Pass-Through	The proxied stream was an on-demand clip, and it was sent in pass-through mode.
Live Pass-Through	The proxied stream was a live clip, and it was sent in pass-through mode.
Live Split	The proxied stream was a live clip, and it was sent using pull-splitting.
Unknown	Clip type and delivery were of an unknown type.

## GET Statements

The GET statement within an access log record shows the path and file name of each file that Helix Proxy served, as well as the protocol and protocol version used to stream or broadcast the file. The following sections show sample entries for GET statements used with different types of on-demand and live content.

**For More Information:** To see the GET statement in context, refer to “Logging Style” on page 158.

### On-Demand Content

The following table lists the formats in which each type of on-demand content is shown in the GET statements of the access log. For a SMIL presentation, a separate record is generated for the SMIL file and for each file in the presentation. When the logging style is set to 5 or 6, you can identify which

files are in the same presentation through the numeric identifier at the end of each access record.

#### GET Statements for On-Demand Content

Feature	Protocol	Example Statement in Access Log
On-demand streamed content	RTSP	"GET presentation/presentation.rm RTSP/1.0"
SMIL files (1 record for the SMIL file, one record for each file listed within the SMIL file)	RTSP	"GET presentation/presentation.smi" "GET presentation/presentation.rt" "GET presentation/presentation.rp" "GET presentation/presentation.rm"
Helix Administrator activity	HTTP	"GET admin/index.html HTTP/1.0"
Authenticated on-demand streamed content	RTSP	"GET secure/topsecret.rm RTSP/1.0"

#### Live Broadcasts

The following table summarizes the format in which each type of live content is shown in the access log.

#### Sample GET Statements for Live Content

Feature	Protocol	Example Statement in Access Log
Unicast, redundant content	RTSP	"GET redundant/live.rm RTSP/1.0"
Unicast content, from RealProducer G2 through 8.5	RTSP	"GET broadcast/live.rm RTSP/1.0"
Unicast content, from pre-G2 encoding source	RTSP	"GET broadcast/live.rm RTSP/1.0"
SLTA content	any	same as live unicast content
Authenticated live streamed content	RTSP	"GET secure/broadcast/live.rm RTSP/1.0"
Multicasting—back-channel	RTSP	"GET broadcast/live.rm RTSP/1.0"

## Client Statistics

All logging styles can include client statistics, which are shown in the preceding sections as [client\_stats\_results]. There are four types of statistics, and the access log can record any combination of them. Each set of statistics is enclosed in square brackets, and begins with a prefix such as Stat1. If you log all four types of statistics, for example, the [client\_stats\_results] field looks like this:

```
[Stat1:statistics_1][Stat2:statistics_2][Stat3:statistics_3][Stat4:statistics_4]
```

Note that although other access log fields are separated by spaces, there is no space between the closing square bracket of one statistics type and the opening square bracket of the next statistics type. The following example shows logging style 5 (see page 160) collecting statistics type 1:

```
207.188.7.125 - - [26/Jun/2008:10:11:03 -0700] "GET real9video.rm RTSP/1.0"
200 858636 [WinNT_5.0_6.0.10.714_RealPlayer_RN92PD_en_686]
[00000000-0000-0000-0000-000000000000] [Stat1: 487 2 1 2 0
44_kbps_Stereo_Music_High_Response_-_RA8] 926322 217 205 1 0 124
[Demand Cache Hit]
```

The following sections describe the information gathered by each of the four statistics types. Statistics 1 and 2 report basic information about playback. Statistics 3 provides information about viewer actions. Statistics 4, which is the default, reports advanced playback information from RealPlayer. The following table lists the media players and versions that can send the various statistics types.

**Media Players and Supported Client Statistics Types**

Media Player	Statistics 1	Statistics 2	Statistics 3	Statistics 4
RealPlayer 2 and earlier	<b>yes</b>	no	no	no
RealPlayer 3 and later	<b>yes</b>	<b>yes</b>	no	no
RealPlayer 5 and later	<b>yes</b>	<b>yes</b>	<b>yes</b>	no
RealOne Player and later	<b>yes</b>	<b>yes</b>	<b>yes</b>	<b>yes</b>
Windows Media Player	<b>limited</b>	<b>limited</b>	<b>yes</b>	no
QuickTime Player	no	no	no	no

Note the following about client statistics:

- Stat1 and Stat2 report codec information only about the audio portion of a clip.

- As noted in the following sections, some statistics are not collected for Windows Media Player. In each case, 0 is typically entered for that statistic.
- Helix Proxy does not record client statistics for QuickTime Player. For each statistics type, [UNKNOWN] is logged.
- Users can choose not to send client statistics. On RealPlayer, the command is **Tools>Preferences> Connection>Internet Settings**. If users select this option, [UNKNOWN] appears in place of that statistics field.
- The statistics interval, described in “Customizing the Access and Error Logs” on page 179, affects how often statistics are reported.

## Statistics Type 1

Statistics type 1 gathers basic information about the success of media clips received by the client. It also tells what codec the client used to decode the audio portion of the clip. For RealNetworks media players, these statistics apply only to a clip’s audio stream. The fields are the following:

[Stat1: received out\_of\_order missing early late codec]

These fields provide the following information:

received	Total number of packets received by the client.
out_of_order	Number of packets received by the client out of order. These packets are reordered as the client plays the clip. This information is not recorded for Windows Media Player.
missing	Number of packets that the client requested, but that did not arrive.
early	Number of requested packets received early by the client. This information is not recorded for Windows Media Player.
late	Number of packets received too late by the client. This information is not recorded for Windows Media Player.
codec	For Windows Media Player, the names of the audio and video codecs used. For RealNetworks clients, the name of the audio codec used to encode the soundtrack. Possible values for RealNetworks players include: s1pr—RealAudio version 5 formats dnet—RealAudio version 3 formats 28.8—RealAudio version 2, 28.8 format lpcJ—RealAudio version 2, 14.4 format cook—RealAudio version 6 format

## Statistics Type 2

Statistics type 2 provides details about the success of clip delivery, giving information about bandwidth requests. Resent packets are described in detail. This statistics type identifies which transport type was used to make the connection, and which audio codec played the clip. For RealNetworks media players, these statistics apply only to a clip's audio stream. This set of statistics uses the following format:

[Stat2: bandwidth available highest lowest average requested received late rebuffering transport startup codec]

The fields provide the following information:

bandwidth	Clip bandwidth in bits per second.
available	Average bits per second available to the user while the clip was playing. This information is not recorded for Windows Media Player.
highest	Highest time between the client resend packet request and the packet resend arrival, in milliseconds. This information is not recorded for Windows Media Player.
lowest	Lowest time between the client resend packet request and the packet resend arrival, in milliseconds. This information is not recorded for Windows Media Player.
average	Average time between the client resend packet request and the packet resend arrival, in milliseconds. This information is not recorded for Windows Media Player.
requested	Number of resend packets requested by the client.
received	Total number of resent packets received by the client.
late	Number of resent packets received by the client too late.
rebuffering	Rebuffering percentage for the clip.
transport	Transport type for the connection. Values are: 0—UDP 1—TCP 2—IP Multicast

startup	Time after the media request that the client received the first clip data package, in milliseconds. The data may arrive before the clip starts playing.
codec	For Windows Media Player, the names of the audio and video codecs used. For RealNetworks clients, the name of the audio codec used to encoded the soundtrack. Possible values include: s1pr—RealAudio version 5 formats dnet—RealAudio version 3 formats 28.8—RealAudio version 2, 28.8 format lpcJ—RealAudio version 2, 14.4 format cook—RealAudio version 6 format

### Statistics Type 3

Statistics type 3 provides detailed information about viewer action while playing clips, but not while receiving live broadcasts. It addresses advanced streaming features, notably ads and image maps. For example, you can find out when a viewer clicked on an image map or stopped the clip. Because each user may carry out several actions, the access log file may grow rapidly when you collect these statistics. Be sure to review the log file frequently, or set up log file rolling to keep the logs to a manageable size. This statistics type uses the following format:

```
[Stat3:timestamp|elapsed_time|action|;]
```

Records of activity are separated by a semicolon (;). Thus, the Stat3 record of a viewer pausing, resuming play, and watching to the clip's end looks like the following:

```
[Stat3:4360|2107|PAUSE|;8401|2107|RESUME|;12608|6321|STOP|;]
```

#### Timestamp

The initial timestamp field gives the time in milliseconds when the action occurred. It is relative to the connection time of the client. In the preceding example, the first timestamp is 4360, meaning the action occurred at 4.360 seconds after the client connected.

#### Elapsed Time

The elapsed\_time field records how many milliseconds into the clip timeline that the action occurred. In the preceding example, the PAUSE action occurs at 2107, or 2.107 seconds into the clip timeline. Notice that the RESUME action

also lists the same elapsed time because this action restarts the clip at the same point where it paused.

### Action

The action field records one of several different actions such as STOP or PAUSE, as described below.

#### CLICK

Viewer clicked on the image map. Further information includes:

x-coord Horizontal coordinate of the click.

y-coord Vertical coordinate of the click.

action Action that occurred. This is one of the following:

PLAYER="*url*"—The URL of a media link the viewer clicked.

URL="*url*"—The URL of a browser link the viewer clicked.

SEEK="*destination*"—The seek destination point, in milliseconds.

#### PAUSE

The viewer paused the client.

#### RESUME

Resume play after a pause, seek, or stop.

#### SEEK

The seek destination point, in milliseconds.

#### STOP

End of clip reached.

#### RECSTART

Media player began recording the clip.

#### RECEMEND

Media player stopped recording the clip.

## Statistics Type 4

Sent only from RealOne Player and later, statistics type 4 gathers most of the same information included in statistics type 1 and type 2, adding packet and bandwidth information for each stream, including the visual tracks of video

clips. RealOne Player through RealPlayer 10 use the following format for statistics type 4:

```
[Stat4:stream_number|mime_type|codec|received|lost|resent|average_bandwidth|current_bandwidth|...information for next stream...|transport turobplay duration clip_end]
```

The following is an example type 4 log entry for a RealVideo Clip sent by RealOne Player through RealPlayer 10:

```
[Stat4:2 audio/x-pn-realaudio|44_kbps_Stereo_Music_High_Response_-_RA8|44100|940|0|0|0|;video/x-pn-realvideo|N/A|180889|2918|0|0| 1 0|1|0| 90 2]
```

RealPlayer 11 and later report additional fields to Helix Proxy 11 and later. These extra statistics are useful for determining media player start-up times and calculating end-to-end latency in live broadcasts. The following shows the format for type 4 statistics reported by RealPlayer 11 and later. The additional fields are shown in **bold**:

```
[Stat4:stream_number|mime_type|codec|received|lost|resent|average_bandwidth|current_bandwidth|...information for next stream...|transport turobplay duration clip_end] startup_time play_time rebuffering_time average_latency|minimum_latency|maximum_latency]
```

The following is an example type 4 log entry for a RealVideo Clip sent by RealOne Player through RealPlayer 10:

```
[Stat4:2 audio/x-pn-realaudio|44_kbps_Stereo_Music_High_Response_-_RA8|44100|940|0|0|0|;video/x-pn-realvideo|N/A|180889|2918|0|0| 1 0|1|0| 90 2] 1228 754100 0 0|0|0
```

### Stream Number

The `stream_number` field indicates how many media streams the clip contains. A video clip might have two streams, for example, one for the audio track and one for the visual track. Following this, information for each stream is reported.

### Stream Information

Helix Proxy reports information for each stream. Information ends with a semicolon. For each stream, the following fields are reported:

<code>mime_type</code>	MIME type, such as <code>audio/x-pn-realaudio</code> .
<code>codec</code>	Codec used for the stream, such as <code>44_kbps_Stereo_Music_High_Response_-_RA8</code> .
<code>received</code>	Number of packets received.

lost	Number of packets lost.
resent	Number of packets resent.
average_bandwidth	Average bandwidth over the course of clip playback in bits per second.
current_bandwidth	The bandwidth in bits per second used when the statistics are reported.

### Transport

The transport field indicates the transport protocol used for the connection. Values are:

0	IP Multicast
1	UDP
2	TCP
3	HTTP cloaked

### TurboPlay

Three turboplay fields indicate the use and results of the RealPlayer TurboPlay feature. The three fields are separated by pipes, as shown here:

```
1|513234|1120
```

The following table lists the possible field values. Values for the second and third field vary depending on whether TurboPlay is on or off, as indicated in the first field.

**TurboPlay Field Values**

Field 1	Field 2	Field 3
0 (off)	Reason TurboPlay is off: 1—User preference. 2—Available bandwidth below 256 Kbps. 3—SureStream in use. 4—Excess rebuffering. 5—Presentation not enabled for TurboPlay. 6—Server not enabled for TurboPlay. 7—Live presentation not supported.	0 (not used)
1 (on)	Accelerated delivery rate in bits per second requested by TurboPlay.	Average buffering time in milliseconds for start of playback, seeking, and so on.

### Duration

The duration field gives the time in milliseconds between the initial client request and the first data packet received by the client.

### Clip End

The clip\_end field lists the reason the presentation ended. Possible values are:

- 0 end of presentation reached
- 1 stop command issued
- 2 reconnection required
- 3 redirection
- PNR\_*n* error code *n* occurred

### Startup Time

Reported by RealPlayer 11 and later, the startup\_time field indicates the time in milliseconds from the initiation of the media request to the point when media begins to play on the viewer's computer.

### Play Time

The play\_time field records the total time in milliseconds that the media player played the streamed media. This excludes the initial buffering time, any rebuffering time, and viewer-initiated pausing. Only RealPlayer 11 and later report this statistics.

### Rebuffering Time

For rebuffering\_time, RealPlayer 11 and later report the cumulative time in milliseconds spent rebuffering the stream.

### Latency Statistics

The three latency values provide information about how quickly RealPlayer rendered data for a live broadcast or a simulated, live broadcast using SLTA. Values are in milliseconds and are reported only by RealPlayer 11 and later.

For a prerecorded, on-demand clip, all three fields always record the value 0. The three fields hold the following information:

average_latency	Indicates the average time to render a data packet once it has been received by RealPlayer.
minimum_latency	Indicates the fastest rendering of a data packet received by RealPlayer.
maximum_latency	Indicates the slowest rendering of a data packet received by RealPlayer.

**Note:** These fields report live broadcast latency on RealPlayer only. Measuring full broadcast latency requires measuring latency introduced by RealProducer, Helix Server, and the network. For more information, refer to the broadcast chapter in *RealProducer User's Guide*.

## Customizing the Access and Error Logs

The following sections explain how to set up basic access and error logging. These logging templates are turned on by default. You may want to change certain default options, however. You can turn off the log files generated through these templates, but you cannot delete the templates.

### Modifying the Basic Access Log

The basic access log is preconfigured to gather basic client statistics for media player requests and write them to a text file. You may want to change the logging style and client statistics types, as well as set up log file rolling.

► **To modify access logging:**

1. Click **Logging & Monitoring>Basic Logging**.
2. For **Logging Style**, choose a number from 0 to 6. The default is 5. For information about the logging style, see “Logging Style” on page 158.
3. If you do not wish to collect client identifiers, choose **Yes** from the **Disable Client GUIDs** pull-down list. This eliminates the collection of client global identifiers, as well as cookie-based IDs. For more information, see “Client Identifier” on page 164.

**Tip:** Cookie-based IDs are also disabled on Helix Proxy if you choose **Yes** for **Disable Client GUIDs**.

4. The **Domain Cookie ID** pull-down list is set to Enabled by default. This means that Helix Proxy attempts to set a cookie on each client. This cookie provides a client ID logged in place of a suppressed GUID when the client requests content. To disable cookie setting, select Disabled. For more information, see “Client Identifier” on page 164.
5. In the **Client Stats** check boxes, select the types of client statistics that each media player reports. You can choose any combination of statistics, or deselect all boxes to gather no client statistics. The default settings are Type 1 and Type 2. For more information, see “Client Statistics” on page 171.

**Tip:** If you gather statistics type 3 or 4, the access log file size will grow rapidly. In this case, be sure to review the log file frequently, or use log file rolling.

6. You can choose to create a new log file at certain intervals, as described in “Log File Rolling” on page 157.
  - a. To create a new log file at regular intervals, set the period through the **Log Rolling Frequency** pull-down lists. You can roll the log hourly, daily, weekly, or monthly.
  - b. To limit the log file by size, type the maximum number of Megabytes in the **Log Rolling Size** box.

**Tip:** Generally, you limit log files by frequency or size. You can select both methods, however, to create log files according to the first limit reached. For example, you can create a new log file whenever the preceding file reaches 10 Megabytes in size, or has recorded 3 days of activity, whichever comes first.

7. For **Access Log File**, you can specify the path and file name for the log file. If you leave this field blank, Helix Proxy records access information in a file named proxy.log in the Logs subdirectory of the Helix Proxy main directory.
8. Click **Apply**.

## Modifying the Basic Error Log

The basic error log captures error information and writes it to a text file. It requires no configuration and cannot be turned off. You may want to set up

log file rolling, though, or specify a different location and name for the log file. For information about the error log syntax, see “Basic Error Log” on page 156.

► To modify the basic error log:

1. Click **Logging & Monitoring>Basic Logging**.
2. You can choose to generate a new error log file at certain intervals, as described in “Log File Rolling” on page 157.
  - a. To create a new log file at regular intervals, set the period through the **Log Rolling Frequency** pull-down lists. You can roll the log hourly, daily, weekly, or monthly.
  - b. To limit the log file by size, type the maximum number of Megabytes in the **Log Rolling Size** box.

**Tip:** Generally, you limit log files by frequency or size. You can select both methods, however, to create log files according to the first limit reached. For example, you can create a new log file whenever the preceding file reaches 10 Megabytes in size, or has recorded 3 days of activity, whichever comes first.
3. In the **Error Log File** field, you can specify the log file name, along with a relative or absolute path. Leave this field blank to use the default path of the Logs subdirectory under the main Helix Proxy directory. The default file name is proxyerr.log.
4. On Windows NT/2000/XP systems, you can send informational and error messages to the Windows Event Viewer. For **NT Event Log Filter**, select the NT error level you want to assign to Helix Proxy error messages.
5. Click **Apply**.



## ADVANCED LOGGING

The advanced logging feature allows you to monitor specific types of events and information that occur on Helix Proxy. You can use this feature to create reports about any type of activity you choose. This chapter explains how to use the advanced logging templates.

### Understanding Advanced Logging

The highly flexible advanced logging feature allows you to gather the exact information you want, reporting it at any time to different outputs such as the screen or a text file. You can use this feature to gather information about current Helix Proxy client connections, for example. Logging templates define the information and format for a log report.

**Tip:** The basic access and error logs, which Chapter 12 describes, are easy to set up and capture a great range of information. You may find them easier to use than defining your own logging templates as described in this chapter.

### The Helix Proxy Registry

To get information for reports, advanced logging relies on information stored in the Helix Proxy registry, which is distinct from the main registry on Windows operating systems. The registry contains information about most aspects of Helix Proxy. Although the registry is an extension of Helix Administrator, there is no link to it from any Helix Administrator page. However, you can display the registry by opening the following URL in a browser:

`http://address:AdminPort/admin/regview.html`

## Registry Variables

The Helix Proxy registry stores information in variables such as `LiveConnections.Count`. Each variable reports a specific type of value or setting, from real-time data on client connections and proxy health, to configuration and license information. When you create an advanced logging template, you add variables to your report by selecting them from a pop-up HTML list. Within a report template, variables are always preceded and followed by percent signs, as in `%LiveConnections.Count%`.

## Global Variables

Through the variables list, you can also choose global variables, such as the time of day, that are derived from the operating system rather than extracted from the Helix Proxy registry. The following table lists the global variables that you can include in reports.

Global Variables	
Variable	Description
<code>%Date%</code>	Indicates the current date in the format MM/DD/YY.
<code>%Time%</code>	Provides the current time of day in the local time zone in the format HH:MM:SS.
<code>%GMTIME%</code>	Displays the current Greenwich Mean Time in the format HH:MM:SS.
<code>%TZDiff%</code>	Indicates the difference between local time and Greenwich Mean Time. For example, the output for Pacific Standard Time is -0800.
<code>%Hour%</code>	Displays the current hour by local time zone in the format HH.
<code>%Min%</code>	Indicates current minute in the format MM.
<code>%Sec%</code>	Adds the current second in the format SS.
<code>%%</code>	Creates a percent sign (%).

## Template Types

You add the registry variables that you want to track to a report template, which defines how often the selected information is reported, as well as where the report is delivered, such as to a file or to the console. You can use four types of templates:

- Interval

Interval templates log information at regular intervals, such as every hour. You can define exactly how much time elapses between report output.

Interval reports are useful for producing regular status reports about Helix Proxy health, for example.

- Watch

With a watch template, you can set a watch on a certain variable, or group of variables, generating a report when information changes. A watch template is useful for reporting errors, for example, because a report is generated only when an error occurs.

- Client Stats

A client statistics template periodically reports statistics from all media player connections. It can generate reports about the number of resent packets and a media players average bit rate, for example. You can generate a report when each media player disconnects, or at periodic intervals, such as every minute.

**For More Information:** The section “Generating Client Statistics Reports” on page 194 provides an example of how to gather client statistics.

- Session

When a system component connects or disconnects, Helix Proxy dynamically adds and deletes variables from its registry. A session template reports on this activity when a component other than a media player (such as a live encoder) connects or disconnects.

**For More Information:** See “Using Session Templates” on page 186 for more information about these templates.

## Report Formats

Through the report template, you format a report, adding boilerplate text around selected variables if you wish. For example, you might create an entry like the following:

With a total of %LiveConnections.Count% player connections, Helix Proxy is using %proxy.ClientBW.Total% bits per second of bandwidth.

In this example, %LiveConnections.Count% and %proxy.ClientBW.Total% are variables, and the rest of the text is boilerplate. When Helix Proxy generates the report, it replaces the variable entries with values from its registry. The resulting report looks like this:

With a total of 50 player connections, Helix Proxy is using 2,800,000 bits per second of bandwidth.

## Using Session Templates

A session template reports on registry variables that are dynamically added and deleted. Helix Proxy creates registry variables when Helix Server and other components connect to it. These variables store information about the component. Using a session template, you can create a report when one of these components connects, disconnects, or both.

**Tip:** To report statistics for each media player, you use a client statistics template instead.

## Choosing a Watch Type

When you create a session template, you select a watch type, which specifies the type of component connection that generates the report. The following table describes the possible values that you can choose.

Watch Types	
Watch Type Value	Registry Values Watched
Broadcast Receiver [BroadcastReceiver.Statistics]	splitting receivers
Broadcast Transmitter [BroadcastDistribution.Statistics]	splitting transmitters
Broadcast [LiveConnections]	live connections
Broadcast Archiver [LiveArchiving.Archiver]	live broadcast archiving
Configuration Change Log [Server.ConfigLog]	configuration file changes

For example, if you choose Configuration Change Log [Server.ConfigLog] as the watch type, you can generate a report every time Helix Administrator updates the Helix Server configuration file. In your report, you then choose which server registry variables you want to log.

**For More Information:** The section “Logging Proxy Configuration Changes” on page 194 provides an example of how to log configuration changes made through Helix Administrator.

## Selecting the Output Format Type

For each session template, you can choose whether to generate the report when the watched component connects, when it disconnects, or both. When you set up the template, you choose an output format from a pull-down list:

- Session Added Output Format    Generate a report with the specified variables when the watched component connects.
- Session Deleted Output Format    Generate a report with the specified variables when the component disconnects.

## Defining Output Methods

The advanced logging output methods determine how Helix Proxy publishes the report. There are several options, and you can select multiple delivery methods for each advanced log report. Additionally, multiple report templates can write to the same output, such as the same file. Most outputs require configuration. For example, if you send your report to a file and a local TCP port, you specify a file name and a port number.

### Console

The Std Error (Standard Error) and Std Out (Standard Output) options both publish the report to the command line console. No configuration is required.

### File

When you select the File output method, Helix Proxy publishes the report to a text file, continuously appending new results to the end of the file unless you set up log rolling. You configure the following variables:

- File name    The log file name. The default location is the main Helix Proxy installation directory. You can specify a relative or absolute path using the syntax appropriate for your operating system.
- Log Rolling Frequency    How many hours, days, weeks, or months pass before a new log file is created (optional).
- Log Rolling Size    Maximum size in Megabytes that the log file can become before a new file is created (optional).

## Using Log File Rolling

Log rolling is optional, but recommended if you expect to report statistics frequently. If multiple templates write to the same file log file, define log rolling in just one template.

### Log Rolling Methods

Generally, you limit log files by frequency or size. You can select both methods, however, to create log files according to the first limit reached. For example, you can create a new log file whenever the preceding file reaches 10 Megabytes in size, or has recorded 3 days of activity, whichever comes first.

### Timestamps

When you implement log rolling, Helix Proxy appends a timestamp to the end of the file name to indicate when the file was created. Suppose that you specify the file name proxystats.txt. Your log directory may contain several files with the same base file name, but each with a unique timestamp that looks like this:

```
proxystats.txt20030622134953
```

The timestamp is in the format YYYYMMDDHHMMSS, using a 24-hour clock. Hence, the file in the preceding example was created on June 22, 2003, at 1:49.53 P.M.

## HTTP Post

With the HTTP Post method, Helix Proxy publishes the report to a Common Gateway Interface (CGI) program. You configure the following variables:

- URL    URL location (excluding http://) of the CGI program. For example:  
        logger.example.com/cgi-bin/report.py
- Port    Number of the HTTP port on the Web server receiving the log.

**Note:** If the attempt to contact the CGI program fails, Helix Proxy writes the message Failed to write log data to HTTP POST Socket to its error log and does not attempt to republish the report. For more information, refer to *Helix Server and Helix Proxy Troubleshooting Guide*.

## TCP Broadcast

The Outbound TCP and Inbound TCP output destinations let you send the report to an application listening on a specific TCP port. The Outbound TCP method publishes the log on a remote computer. For this method, you configure the following variables:

Destination    Host name or IP address of the computer that receives the log.  
 Port            Number of an open port on the specified computer.

The Inbound TCP method publishes the log on the local computer. You configure the following variable:

Port            Number of an open port on the local computer.

## UDP Broadcast

The Outbound UDP and Multicast UDP methods publish the report to a UDP socket on a remote computer using unicast or multicast UDP, respectively. You configure the following variables:

Destination    Host name or IP address of the computer where the report should be published. For Multicast UDP, enter a Class D IP multicast address.  
 Port            Number of an open port on the specified computer.

## Pipe and System Log on UNIX

On UNIX operating systems, the Pipe and Syslog methods make the log available to another process, or publish the information to the system log, respectively. For Pipe, you configure the following variable:

Command        Pipe command to the application or script where the information can be post-processed.

For Syslog, you choose one the following priorities, each of which corresponds to an entry type in the UNIX system log:

- LOG\_EMERG
- LOG\_ALERT
- LOG\_CRIT
- LOG\_ERR

- LOG\_WARNING
- LOG\_NOTICE
- LOG\_INFO
- LOG\_DEBUG

## Windows NT Event Log

If you choose NT Event Log, Helix Proxy publishes the report to the event log that corresponds to the priority selected. Each option corresponds to an entry type in the Windows NT Event Log:

- LOG\_ERR
- LOG\_WARNING
- LOG\_INFO

## Creating Logging Templates

The following procedure explains how to create a new logging template, or modify the predefined templates. You'll need to be familiar with the information in the preceding sections to set up your custom template.

► To create or modify an advanced logging template:

1. Click **Logging & Monitoring**>**Advanced Logging**.
2. To create a new template, choose Interval, Watch, Client Stats, or Session from the **Add Template** pull-down list. The option you choose affects other options that appear on the page, as described in Step 8 through Step 11. To select an existing template, highlight its name in the **Templates** area.

**For More Information:** See “Template Types” on page 184.

3. If you are creating a new template, edit the name in the **Template Name** box. This name is for your reference only.
4. The **Template Type** pull-down list indicates the type of template you chose (Interval, Watch, Client Stats, or Session). You can change the template type here if needed.

5. From the **Template Status** box, select On or Off to enable or disable the logging report, respectively. An existing template starts or stops reporting as soon as you change its status and click **Apply**.
6. Optionally, you can enter a description in the **Template Description** box. This is for your own reference only, but is highly recommended.
7. You next select one or more output types for the report to determine where Helix Proxy sends the report information:
  - a. Select an output type from the **Add Output Type** pull-down list.
  - b. Optionally, edit the name in the **Output Name** box. This name is for your reference only.
  - c. For the selected output type, enter the necessary configuration parameters, as described in “Defining Output Methods” on page 187.
8. If you chose Interval in Step 2, follow this step. Otherwise, skip to the next step. For the Interval template, set the appropriate combination of hour, minute, and seconds for the report interval in the **Output Interval** boxes. If you leave a box blank, the setting for that box is considered to be 0. Next, follow the instructions in Step 12.
9. If you chose a Watch template in Step 2, follow this step. Otherwise, skip to the next step. For the Watch template, click Property List in the **Watches** area. In the list that appears, choose the variable or variables that you want to watch for changes.

The optional minimum and maximum output intervals for the Watch template let you generate the report at regular intervals. If you do not define either field, Helix Proxy creates the report only when a watched registry variable changes:

- **Minimum Output Interval**

This field holds a number in the format HH:MM:SS that defines the smallest possible time that must pass between log outputs. Changes to the watched variables are not reported until the minimum interval has elapsed. If you set 00:05:00, for example, watched variables that change are reported every five minutes. If no watched variable changes its value in five minutes, though, the report is not created until a variable changes, or the maximum interval is reached.

- **Maximum Output Interval**

This entry contains a number in the format HH:MM:SS that defines the longest possible time allowed to pass before the report output is generated. Changes to variables being watched will generate the report output even if the maximum interval has not been reached, however. If you set 01:00:00, for instance, and no watched variable changes within an hour after the last report, Helix Proxy generates the report when the hour elapses.

Next, follow the instructions in Step 12.

**Note:** Place each watched variable on a separate line in the **Watches** list. Helix Proxy determines which variables to watch by matching the string that appears on each line of the **Watches** list.

10. If you chose Client Stats in Step 2, follow this step. Otherwise, skip to the next step. For the Client Stats template, set the appropriate combination of hour, minute, and seconds for the report interval in the **Output Interval** boxes. If you leave a box blank, the setting for that box is considered to be 0. Next, follow the instructions in Step 12.

**Note:** The report intervals are relative to the start of the player session. So if the interval is five minutes and player A connects at 12:00 whereas player B connects at 12:01, the first reports for players A and B are generated at 12:05 and 12:06, respectively.

11. If you chose Session in Step 2, select the appropriate watch type from the **Watch Type** list box. As described in “Choosing a Watch Type” on page 186, the watch type you select determines which dynamic event triggers the report output.
12. For any template type, click **Property List** in the **Output Format** area to pick the variables from the Helix Proxy registry included in the template. Note that you can define two report outputs for some template types:
  - If you’re setting up a Session template, you can specify one output format by selecting **Session Added Output Format**, and a second format by choosing **Session Deleted Output Format**. These output formats for the beginning and end of the watched session can be identical or different.
  - For a Client Stats template, you can choose **Periodic Output Format** to specify an output format generated whenever the **Output Interval** time

elapses. By selecting **Disconnect Output Format**, you can create another output format that is generated whenever a media player stops playing a presentation.

Do the following to define the output format for any template type:

- a. In the property list window, navigate to the variable that you want to include in the template.
- b. When you click on a variable, a string identifying that variable appears in the **Output Format** text box. Helix Proxy reports values for variables in the exact order the variables appear in the text box. To organize the order of variables, cut and paste them in the order that you want them to appear in your report.

**Tip:** A variable added to the **Output Format** text box is surrounded by percentage signs (%Server.Bandwidth.Output%). If you reorganize the order of the variables, be sure to include the percent signs that surround each variable name.

**For More Information:** For descriptions of the registry properties, refer to the registry properties section of *Helix Proxy Configuration and Registry Reference*.

- c. Optionally, format the report output by adding boilerplate text. Two tags help with formatting the output string. Use a `\n` tag to move output to a new line. Carriage returns you enter in the box are also recognized as new lines. Use a `\t` tag to insert a tab.

13. Click **Apply**.

## Sample Templates

This section explains how to use the preconfigured template that comes with Helix Proxy. It also provides examples of setting additional templates to log client statistics and changes to Helix Administrator.

### Using the Server Stats Templates

The preconfigured Server Stats template is an example of an interval template. It is designed to send basic proxy statistics to the output console every hour. To use it, you must enable it and, optionally, customize it by changing the

output destination or modifying the reporting variables. The report output looks like this:

```
Server Stats (06/17/02 10:33:52)
  Uptime: 1234274 seconds
  CPU Percent Usage: 5
  Players Connected: 32
  Players Connected in the Last 10 Seconds: 2
  Players Connected by Protocol: 22 RTSP, 10 MMS, 0 HTTP (0 Cloaked)
  Players Connected by Transport: 0 TCP, 32 UDP, 0 MCast
  Total Subscribed Bandwidth Output: 9385984 bps
  Total Actual Bandwidth Output: 9244432 bps
  Average Bandwidth Output Per Player: 293312 bps
  Memory Stats: 14294824 Bytes In Use
```

## Logging Proxy Configuration Changes

Using a Session template, you can log changes to Helix Proxy made through Helix Administrator. Follow the instructions for setting up a Session template as described in “Creating Logging Templates” on page 190, setting any desired output, such as the screen console or a text file. For **Watch Type**, choose Configuration Change Log [Server.ConfigLog]. For **Session Added Output Format**, enter the following to capture all configuration changes:

```
%Server.ConfigLog.*.Entry%
```

The report indicates the IP address and user name of the person who made the change, along with the date, time, and browser version. It then lists the changes made to the Helix Proxy registry, which are recorded in the configuration file. The following is a sample report entry:

```
127.0.0.1 - FBLACK.AdminRealm/fblack [15/Aug/2003:12:26:28 -0700]
"POST admin/configvar.set.html HTTP/1.0" - - [Mozilla/4.0
(compatible;MSIE 6.0;Windows NT 5.1;.NET CLR 1.0.3705)]
Set config.DiffServ.Control=0 [OK]
Set config.DiffServ.Media=17 [OK]
```

**Tip:** The IP address, date, time, and browser fields are the same as those used in the basic access log. For more information, see “Access Log Fields” on page 161.

## Generating Client Statistics Reports

This example illustrates how to use a Client Stats template to log information about each media player request. This sample template generates periodic

updates about the session status, such as the current number of lost packets. When the player disconnects, the disconnection report provides information about the entire session, such as the total number of packets lost, the requested URL, the streaming protocol, and so on.

### Periodic Client Statistics Report

For **Periodic Output Format**, you define the report information you want to collect for each media player whenever the **Output Interval** time elapses. You create a report using boilerplate text and variables chosen from the pop-up property list. Note that `\n` adds a new line to the report. You can also use `\t` to insert tabs. The following is the sample template definition in Helix Administrator:

```
\n\n****PERIODIC CLIENT STATISTICS****
Date and Time: %Date%, %Hour%:%Min%.%Sec%
Client GUID: %Client.*.GUID%
Average Bit Rate: %Client.*.Session.*.AvgBitrate%
Bytes Sent: %Client.*.Session.*.BytesSent%
Packets Sent: %Client.*.Session.*.PacketsSent%
Packets Lost: %Client.*.Session.*.PacketsLost%
Failed Resends: %Client.*.Session.*.FailedResends%
Successful Resends: %Client.*.Session.*.SuccessfulResends%
```

The following text is an example of a report generated from the preceding template:

```
****PERIODIC CLIENT STATISTICS****
Date and Time: 08/21/03, 15:50.04
Client GUID: 3ab16eb2-9f30-4c1f-acb6-25dfba5ba0da
Average Bit Rate: 225000
Bytes Sent: 1798645
Packets Sent: 588
Packets Lost: 1
Failed Resends: 0
Successful Resends: 1
```

**For More Information:** For details about how Helix Server determines the client GUID, see “Client Identifier” on page 164.

## Disconnection Statistics Report

For **Disconnect Output Format**, you define the report information you want to collect when a client disconnects. Here's the template defined in Helix Administrator:

```
\n\n****CLIENT DISCONNECT****
Date and Time: %Date%, %Hour%:%Min%.%Sec%
**CLIENT INFORMATION**
Client GUID: %Client.*.GUID%
Client ID: %Client.*.ClientID%
Type: %Client.*.User-Agent%
Address: %Client.*.Addr%
Preferred Language: %Client.*.Language%
**CONNECTION STATUS**
Average Bit Rate: %Client.*.Session.*.AvgBitrate%
Bytes Sent: %Client.*.Session.*.BytesSent%
Packets Sent: %Client.*.Session.*.PacketsSent%
Packets Lost: %Client.*.Session.*.PacketsLost%
Failed Resends: %Client.*.Session.*.FailedResends%
Successful Resends: %Client.*.Session.*.SuccessfulResends%
**CLIP INFORMATION**
Requested URL: %Client.*.Session.*.PlayerRequestedURL%
Start Time: %Client.*.StartTime%
Clip Size in Bytes: %Client.*.Session.*.FileSize%
Playing Duration (seconds): %Client.*.Session.*.DurationSeconds%
Title: %Client.*.Session.*.FileHeader.Title%
Author: %Client.*.Session.*.FileHeader.Author%
Copyright: %Client.*.Session.*.FileHeader.Copyright%
Stream Count: %Client.*.Session.*.FileHeader.StreamCount%
**TRANSPORT INFORMATION**
Protocol: %Client.*.Protocol%
Port: %Client.*.Port%
UDP used (0=no, 1=yes): %Client.*.IsUDP%
```

The following is output generated from the preceding template. Helix Proxy generates this report only when a media player stops playing a presentation:

```
****CLIENT DISCONNECT****
Date and Time: 08/21/03, 15:51:00
**CLIENT INFORMATION**
Client GUID: 3ab16eb2-9f30-4c1f-acb6-25dfba5ba0da
Client ID: WinNT_5.0_6.0.11.818_RealPlayer_RN10PD_en-us_686
Type: RealMedia Player Version 6.0.9.1753 (win32)
Address: 207.188.7.125
Preferred Language: en-us
```

```
**CONNECTION STATUS**  
Average Bit Rate: 225000  
Bytes Sent: 2478645  
Packets Sent: 643  
Packets Lost: 1  
Failed Resends: 0  
Successful Resends: 1  
**CLIP INFORMATION**  
Requested URL: rtsp://208.147.89.157:554/video1.rm  
Start Time: 21/Aug/2003:15:48:50  
Clip Size in Bytes: 2479645  
Playing Duration (seconds): 130  
Title: Introductory Video  
Author: RealNetworks, Inc.  
Copyright: ©2002 RealNetworks, Inc.  
Stream Count: 1  
**TRANSPORT INFORMATION**  
Protocol: RTSP  
Port: 7180  
UDP used (0=no, 1=yes): 1
```

**For More Information:** For descriptions of the client registry properties, refer to the client properties chapter of *Helix Proxy Configuration and Registry Reference*.



## PROXY MONITOR

Using the Proxy Monitor, you can monitor inbound and outgoing bandwidth, and view the number of clients currently connected. This chapter explains how to use the Proxy Monitor.

**For More Information:** To generate reports of historical activity, refer to Chapter 12. RSS statistics track proxy activity over time. For more information, refer to the chapter on RSS statistics in *Helix Server and Helix Proxy Troubleshooting Guide*.

### Viewing Helix Proxy Activity

In Helix Administrator, click **Logging & Monitoring>Proxy Monitor**. The monitor page appears in the right-hand frame.

#### Proxy Monitor in Helix Administrator

Proxy Monitor		HELP
Connected Clients		152
Total Clients Served		643
Data Source	Client Traffic	Gateway Traffic
Proxy	68,815,435	145,056
Block Cache Import		647,680
Stream Cache Import		0
Splitter Import		20,335,690
Total	68,815,435	21,128,426

**Tip:** By default, the monitor statistics update every 60 seconds. You can change the update frequency by editing the `RSSInterval` variable in the Helix Proxy configuration file (`rmproxy.cfg`). For details, refer to the chapter on RSS statistics in *Helix Server and Helix Proxy Troubleshooting Guide*.

## Media Player Statistics

The top two lines of the Proxy Monitor show the following statistics:

- **Connected Clients**—The number of media players receiving streams from Helix Proxy when the monitor statistics were last updated.
- **Total Clients Served**—Number of unique media player TCP connections established since Helix Proxy was last started.

**Tip:** The actual number of *unique media players* served may be lower than the Total Clients Served value. Suppose that a user requests one stream, closes the media player, restarts it, then requests another stream. In this case, the second stream counts as a second TCP connection. In this instance, therefore, a single media player was responsible for two control connections that were logged as two clients served.

## Bandwidth Statistics

Beneath the client statistics lines, a table indicates the average incoming and outgoing bandwidth during the current statistics reporting period. The **Client Traffic** column indicates outgoing bandwidth. The **Gateway Traffic** column provides statistics about incoming proxy data. All values are average bits per second.

**Proxy Monitor Bandwidth Values**

Data Source	Client Traffic	Gateway Traffic
Proxy	Outgoing bandwidth used to fulfill on-demand and live stream requests.	Incoming bandwidth for live and on-demand streams that Helix Proxy passes through to media players without caching or splitting. This figure also includes the bandwidth for control connections to origin servers.
Block Cache Import	Not applicable.	Incoming bandwidth for on-demand, RTSP-streamed clips that are being delivered by an origin server and added to the proxy cache.
Stream Cache Import	Not Applicable	Incoming bandwidth for on-demand, MMS-streamed clips that are being delivered by an origin server and added to the proxy cache.

(Table Page 1 of 2)

**Proxy Monitor Bandwidth Values (continued)**

Data Source	Client Traffic	Gateway Traffic
Splitter Import	Not applicable.	Incoming bandwidth of live media streams split by the proxy.
Total	Total outgoing bandwidth.	Total incoming bandwidth.

(Table Page 2 of 2)



## SNMP

Using Simple Network Monitoring Protocol (SNMP), you can monitor Helix Proxy from an SNMP management system. This allows you to change Helix Proxy configuration from a third-party tool, and send notice of important events to an external program. This chapter explains how to set up the SNMP monitoring plug-in and the Helix Proxy master agent.

## Understanding SNMP

The following sections describe the components of the Helix Proxy SNMP monitoring system. Before implementing SNMP on Helix Proxy, be sure that you understand the basics of SNMP monitoring and know how to operate your chosen SNMP management system.

### SNMP Plug-in

Helix Proxy includes an SNMP plug-in that monitors its registry for configuration values and events. The plug-in communicates to the master agent using a proprietary protocol. It can send important information about Helix Proxy operation to the master agent, and update the Helix Proxy configuration as instructed by the management system. You must configure the plug-in before it can operate.

**For More Information:** The section “Configuring the SNMP Plug-In” on page 205 explains how to set up the plug-in. Refer to “License File Information” on page 41 for more about licensed features.

### Master Agent

The SNMP plug-in communicates with the master agent, an executable program included with Helix Proxy. The master agent then communicates

with the management system using the SNMP protocol. The SNMP plug-in and the management system never communicate directly. The master agent can run as an independent application or a Windows service. Once configured, the master agent generally runs without the need for user intervention.

**For More Information:** See “Configuring the Master Agent” on page 207.

## SNMP Protocol

The master agent uses the SNMP protocol to communicate with the management system. It supports SNMP version 1 (SNMPv1), version 2c (SNMPv2c), and version 3 (SNMPv3). Versions 1 and 2 of the SNMP protocol do not encrypt messages between the two components, and are therefore recommended only when both Helix Proxy and the management system reside behind a firewall on the same private network.

**Note:** Helix Proxy does not support SNMP over Internet Protocol version 6 (IPv6). Components must use IPv4 addresses.

### SNMP Version 3 Protocols

SNMPv3 is suitable for communications over an unprotected network. The User-based Security Model (USM) for SNMPv3 defines two authentication protocols, both of which are supported for Helix Proxy SNMP:

- HMAC-MD5-96

This protocol is based on MD5. Operations using MD5 occur faster than those using SHA.

- HMAC-SHA-96

This protocol is based on SHA-1. SHA provides a stronger security mechanism than MD5.

### SNMP Version 3 Security Levels

SNMPv3 defines three levels of security. The lowest level (noAuthNoPriv) does not provide authentication or privacy, and is comparable to SNMP version 1.

The second level (AuthNoPriv) provides authentication, but no privacy. The third level (AuthPriv) provides authentication and encryption for all messages.

#### SNMP Versions and Authentication Modes

SNMP Version	Authentication Mode	Operation
version 1 (SNMPv1)	noAuthNoPriv	Authentication is performed by matching an unencrypted community string. This method is not suitable for communication across an unsecured network.
version 2 (SNMPv2c)	noAuthNoPriv	
version 3 (SNMPv3)	noAuthNoPriv	This mode provides authentication based on the HMAC-MD5 or HMAC-SHA algorithm.
	authPriv	

## Management System and Management Information Base (MIB)

You can use any third-party SNMP monitoring tool as your management system. The management information base (MIB) determines the Helix Proxy configuration variables that the management system monitors and controls. It also defines the event traps that the SNMP plug-in can report to the master agent. Helix Proxy ships with a MIB configuration file named `helixserver.my`, located in the main Helix Proxy installation directory.

**For More Information:** The section “Running a Management System” on page 214 explains the monitoring trees that appear in the management system.

## Configuring the SNMP Plug-In

You configure the SNMP plug-in through Helix Administrator. The configuration connects the plug-in to the master agent, and defines which events to report to the management system.

► To configure the SNMP plug-in:

1. Click **Logging & Monitoring>SNMP**.
2. To use SNMP monitoring, set the value of **Enable SNMP** to Yes.

3. In the **Master Agent Hostname or IP Address**, enter the DNS name or IPv4 address used by the master agent program. The agent typically resides on the Helix Proxy machine in the Helix Proxy Bin directory, so you can use the localhost address of 127.0.0.1.
4. For **Master Agent Port**, enter the master agent port used for the AgentX protocol. The default is port 705.

**Note:** This value must match the value for the AgentXProtocolPort variable entered in the master agent configuration file. See “Configuring the Master Agent” on page 207 for more information.

5. For **Send SNMP Traps**, select Yes if you want the SNMP plug-in to notify the management system when specific events occur, such as CPU utilization above a certain percentage. If you leave the pull-down list set to No, the SNMP plug-in responds to management system requests, but does not report Helix Proxy events. In this case, you can ignore the remaining fields, which define the event values to trap, and click **Apply** to save your changes.

**Tip:** You can disable an individual trap by setting its value to 0.

6. The **Trap Interval** field determines how frequently a trap is triggered while the trap condition remains active. The value is in seconds, with 20 as the default. For example, if you trap CPU utilization, the trap triggers every 20 seconds as long as the CPU stays above the specified level of use.
7. The **Trap Server Startups** pull-down determines if the SNMP plug-in notifies the management system of Helix Proxy restarts. Change the value to No if you do not want to set this trap.
8. In the **Trap CPU Utilization Above** field, enter an integer in the range from 0 to 100 that indicates the percentage of total machine CPU utilization that triggers a trap. A value of 75, for example, initiates a trap when CPU utilization on the Helix Proxy machine reaches 75 percent or higher.

**Tip:** CPU utilization is reported by the operating system. On multiprocessor machines, this figure represents the aggregate load for all processors.

9. For **Trap Connection Counts Above** field, enter an integer that indicates a number of simultaneous media player connections that triggers a trap. A

value of 1000, for example, initiates a trap when 1,000 or more media streams are being delivered.

10. The three fields for **Memory Usage Watermarks** allow you to set up to three traps reported when Helix Proxy memory usage rises above a specified amount. Use an integer value that represents Kilobytes. If you set 128000, 180000, and 230000, for example, the SNMP plug-in reports memory usage that exceeds approximately 128, 180, and 230 Megabytes, respectively.

**Tip:** Set the traps at 50, 70, and 90 percent, respectively, of memory allotted to Helix Proxy. The values shown above represent these percentages on a machine in which Helix Proxy has a dedicated allotment of 256 Megabytes.

11. For **Bandwidth Usage Watermarks**, you can define up to three traps that trigger when the Helix Proxy outgoing bandwidth use rises above the amount specified in Kilobits per second (Kbps). Each field accepts an integer value. If you set 2000, 4000, and 8000, for example, the SNMP plug-in reports when outgoing bandwidth exceeds approximately 2, 4, and 8 Megabits per second, respectively.
12. Click **Apply**.

## Configuring the Master Agent

The master agent is the intermediary through which the SNMP plug-in and the management system communicate. It must always run on the Helix Proxy machine. The following sections explain how to modify the master agent configuration file to define your system addresses, users, and security model.

### Modifying the Master Agent Configuration File

You use the `master.cfg` file installed in the Helix Proxy installation directory to configure the master agent. This allows the master agent to communicate with the SNMP plug-in and the management system. It also defines the security level for each person who uses the management system. Edit this XML-formatted text file using any text, HTML, or XML editor. The following example shows the default configuration file:

```

<?xml version="1.0" encoding="US-ASCII"?>
<preferences version="0.5">
  <config ManagerAddress="127.0.0.1" ManagerSNMPPort="162"
    LocalSNMPPort="161" AgentXProtocolPort="705" EngineID="XXX"/>
  <security CommunityString="public"/>
  <SecurityModel ModelType="USM">
    <users UserName="xxx">
      <Authentication Type="MD5" Password="yyy"/>
      <Privacy Type="DES" Password="zzz"/>
    </users>
    <users UserName="unsecureUser">
      <Authentication Type="NONE" Password=""/>
      <Privacy Type="NONE" Password=""/>
    </users>
  </SecurityModel>

  <SecurityToGroup SecurityModel="USM" User="unsecureUser" Group="v3Group"/>
  <SecurityToGroup SecurityModel="USM" User="test" Group="testGroup"/>
  <SecurityToGroup SecurityModel="v2" User="vishal" Group="v1v2group"/>
  <SecurityToGroup SecurityModel="v1" User="public" Group="v1v2group"/>

  <SecurityModel ModelType="VACM">
    <groups Name="v3Group" SecurityModel="USM" SecurityLevel="1"
      Context="" Notify_View="testView" Read_View="testView"
      Write_View="testView"/>
    <groups Name="testGroup" SecurityModel="USM" SecurityLevel="1"
      Context="" Notify_View="testView" Read_View="testView"
      Write_View="testView"/>
    <groups Name="v1v2group" SecurityModel="v1" SecurityLevel="1"
      Context="" Notify_View="v1NotifyView" Read_View="v1ReadView"
      Write_View="v1WriteView"/>
    <groups Name="v1v2group" SecurityModel="v2" SecurityLevel="1"
      Context="" Notify_View="v1NotifyView" Read_View="v1ReadView"
      Write_View="v1WriteView"/>

    <views Name="testView" OID="1.3" Mask="" Included="1"/>
    <views Name="v1ReadView" OID="1.3" Mask="" Included="1"/>
    <views Name="v1WriteView" OID="1.3" Mask="" Included="1"/>
    <views Name="v1NotifyView" OID="1.3" Mask="" Included="1"/>

  </SecurityModel>
</preferences>

```

## Defining Master Agent Addresses and Ports

The following lines in the master agent configuration define the basic communication between the master agent, the SNMP plug-in, and the management system:

```
<config ManagerAddress="127.0.0.1" ManagerSNMPPort="162"
  LocalSNMPPort="161" AgentXProtocolPort="705" EngineID="XXX"/>
```

The following table explains the values you should set for these attributes.

Master Agent Address and Port Attributes	
Attribute	Value
ManagerAddress	The IPv4 address of the management system. The master agent uses this address to send traps to the management system. You must specify an IP address, not a DNS name. The master agent does not support IPv6 addresses.
ManagerSNMPPort	The port used by the management system to listen for communication from the master agent. The default is port 162.
LocalSNMPPort	The local port used by the master agent for SNMP communications. The default is port 161. If another application uses an SNMP process, port 161 may be in use. In this case, specify a free port. Do not use port 162, as this can cause a system slowdown.
AgentXProtocolPort	The master agent port for AgentX, which is the protocol used to communicate with the Helix Proxy SNMP plug-in. The default is 705.

## Setting Up SNMP Security

The following lines set the parameters for SNMP security. The USM security model defines the access rights for each person running the management system. The configuration file predefines two users. The first user operates with no security, which is equivalent to using SNMP version 1. The second user defines authentication and privacy, the highest security under SNMPv3. You can modify or delete these predefined users, as well as create additional users by adding new `<users>...</users>` lists within the USM section:

```
<security CommunityString="public"/>
<SecurityModel ModelType="USM">
  <users UserName="xxx">
    <Authentication Type="MD5" Password="yyy"/>
```

```

    <Privacy Type="DES" Password="zzz"/>
  </users>
  <users UserName="unsecureUser">
    <Authentication Type="NONE" Password="" />
    <Privacy Type="NONE" Password="" />
  </users>
</SecurityModel>

```

The following table defines the master agent configuration attributes that define the SNMP security level and permissions.

<b>Master Agent Address and Port Attributes</b>	
Attribute	Value
CommunityString	Password used with SNMP version 1 or 2. You can ignore the <security/> tag if you are using SNMP version 3. If you are using version 1 or 2, you can ignore the USM settings.
UserName	Name of the user as defined in the management system.
Authentication Type	Type of authentication used with SNMPv3. Valid values are MD5 for the HMAC-MD5 algorithm, or SHA for the HMAC-SHA algorithm. A value of NONE indicates an unsecured user.
Privacy Type	For privacy type, you can enter NONE for no privacy or DES for CBC-DES encryption.
Password	Password for authentication or privacy. SNMPv3 uses separate passwords for authentication and privacy. You do not need to define a certain password, however, if you used NONE as the authentication or privacy type.

## Defining a View Access Control Model

The view access control model (VACM) available through SNMPv3 allows you to define precisely which Helix Proxy SNMP objects each viewer can see and control. VACM is optional, and you should be familiar with how it works within your SNMP management system before you define view privileges through the master agent configuration file. The following sections provide an example of how to set up view access for a specific user.

### Assigning a User to a Group

Each person who uses VACM must be defined in the <SecurityModel> list as an SNMPv3 user. The following example shows a user defined to use SNMPv3 with authentication but no privacy:

```

<SecurityModel ModelType="USM">
  <users UserName="Maria">
    <Authentication Type="MD5" Password="tl73jkIL98"/>
    <Privacy Type="NONE" Password=" "/>
  </users>
  ...other users defined here...
</SecurityModel>

```

Using a `<SecurityToGroup/>` tag, you assign each user to a group name that you create. In the following example, the user Maria is assigned to a group named `v3Group`:

```
<SecurityToGroup SecurityModel="USM" User="Maria" Group="v3Group"/>
```

The following table explains the `<SecurityToGroup/>` tag attributes.

**VACM `<SecurityToGroup/>` Tag Attributes**

Attribute	Value
SecurityModel	Security model for this user. Choose one of the following: <ul style="list-style-type: none"> <li>- v1 for SNMP version 1</li> <li>- v2c for SNMP version 2c</li> <li>- USM for SNMP version 3</li> </ul>
User	The user's name.
Group	The group to which the user is assigned. Groups are defined with <code>&lt;groups/&gt;</code> tags.

## Creating Groups

Within the `<SecurityModel>` list, a `<groups/>` tag defines each group. A group has three views, indicating which parts of Helix Proxy the user can monitor and control. In the following example, `v3Group` is assigned the `fullView` view for receiving traps and reading Helix Proxy variables. It is part of the `noView` view for writing configuration changes to the Helix Proxy registry:

```

<SecurityModel ModelType="VACM">
  <groups Name="v3Group" SecurityModel="USM" SecurityLevel="1"
    Context="" Notify_View="fullView" Read_View="fullView"
    Write_View="noView"/>
  ...more groups and views defined here...
</SecurityModel>

```

The following table explains the <groups/> tag attributes.

**VACM <groups/> Tag Attributes**

Attribute	Value
Name	The group name. Users are assigned to this group by including the name in the <SecurityToGroup/> tag.
SecurityModel	Security model for this group. Choose one of the following: - v1 for SNMP version 1 - v2c for SNMP version 2c - USM for SNMP version 3
SecurityLevel	Security level for this group. Choose one of the following: - 0 for noAuthNoPriv - 1 for authNoPriv - 2 for authPriv
Context	An optional, named subset of object instances in the management information base.
Notify_View	The name of the view assigned to the group for receiving traps.
Read_View	The name of the view assigned to the group for reading SNMP objects values corresponding to Helix Proxy registry values.
Write_View	The name of the view assigned to the group for writing changes to SNMP object values and thereby changing Helix Proxy configuration values.

## Defining Views

Within the <SecurityModel> list, a <views/> tag defines each view. The view identifies a group of objects by an OID from the management information base (MIB). All objects that fall under that OID are included in the view. In the following example, the fullView view is included while the noView view is excluded, allowing no access:

```
<SecurityModel ModelType="VACM">
  ...groups defined here...
  <views Name="fullView" OID="1.3" Mask="" Included="1"/>
  <views Name="noView" OID="1.3" Mask="" Included="0"/>
  ...more views defined here...
</SecurityModel>
```

The following table explains the <views/> tag attributes.

**VACM <views/> Tag Attributes**

Attribute	Value
Name	The view name. Groups are assigned up to three views (Notify_View, Read_View, and Write_View) in each <groups/> tag.
OID	Object ID of a node. All objects that fall under that node in a tree are available in the view. The MIB file used by the SNMP management system lists the OIDs of all nodes.
Mask	Optional mask value that applies to the OID. You can use this mask to provide finer control over the objects available in the view.
Included	A true or false value that includes or excludes the view. Use 1 to make the view available, 0 to turn exclude the view from use.

## Running the Master Agent on Windows

On Windows, you can run the master agent as a service or as an application. The following sections explain how to start the master agent in either mode.

### Restarting the Master Agent Service

If you installed the master agent as a Windows Service as described in “Installing Helix Proxy” on page 30, the agent starts up automatically. If you change the master agent configuration, restart the agent service by locating the master agent service with **Settings>Control Panel>Administrative Tools>Services**. In the **Services** dialog, right-click on **SNMP Master Agent** and choose **Restart**.

**Tip:** Right-click **SNMP Master Agent** and choose **Properties** to change the master agent operation. Using this dialog, for example, you can disable automatic start-up or restart the service automatically if it fails.

### Starting the Master Agent as an Application

The following procedure explains how to start the master agent as a Windows application. Do this only if the master agent has not been installed as a service, or you have disabled the service through the **Services** dialog.

- Starting the master agent as a Windows program:
  1. Open a command prompt using **Start>Program>Accessories>Command Prompt**.

2. Navigate to the Helix Proxy installation directory. For example:  
`cd "C:\Program Files\Real\Helix Proxy"`
3. The master agent uses the executable name `master.exe` and resides in the `Bin` subdirectory. Start it by entering the following:  
`Bin\master.exe master.cfg`
4. Start Helix Proxy as described in “Starting Helix Proxy” on page 33.

## Starting the Master Agent on UNIX

The following procedure explains how to start the master agent as a UNIX background process.

► **To start the master agent on UNIX:**

1. If Helix Proxy is running, shut it down as described in “Stopping Helix Proxy” on page 36.
2. Log in as root.
3. From the command line, navigate to the Helix Proxy installation directory. For example:  
`# cd /usr/local/Real/HelixProxy`
4. The master agent uses the executable name `master` and resides in the `Bin` subdirectory. Start it as a background process:  
`# ./Bin/master master.cfg &`
5. Start Helix Proxy as described in “Starting Helix Proxy” on page 33.

## Running a Management System

Once you have the SNMP plug-in and master agent configured and running, you can use your management system to monitor and control Helix Proxy. From the management system, locate the MIB file, which is named `helixserver.my` and resides in the Helix Proxy installation directory. Compile the MIB file if necessary for your management system. You then connect the management system to the master agent using the Helix Proxy IP address and port defined in the master agent configuration file.

**For More Information:** For information about compiling the MIB file and connecting your management system to the master agent, refer to your SNMP manager documentation.

The section “Defining Master Agent Addresses and Ports” on page 209 explains the master agent port usage.

## Monitor Tree

The MIB file produces two main trees in the SNMP management system. Through these monitoring trees, you can monitor Helix Proxy operation and control the settings of certain variables. The hpMonitor tree contains objects related to Helix Proxy monitoring. These objects, described in the following table, cannot be changed by the management system.

<b>Monitor Tree Objects</b>	
Object	Information
hpPseudonym	Helix Proxy unique identifier.
hpPassthruStreams	Current number of live and on-demand streams being delivered in pass-through mode.
hpCachedStreams	Number of streams being served from the Helix Proxy cache.
hpSplitStreams	Number of live streams being split.
hpTotalConnections	Total number of connections
hpClientBandwidthTotal	Total bandwidth for client streams being served.
hpProxyBandwidthTotal	Sum of proxy bandwidth.
hpRTSPCacheImportTotal	Import bandwidth used by RTSP clients.
hpMMSCacheImportTotal	Import bandwidth used by MMS clients.
hpSplitterGatewayBWTotal	Sum of splitter gateway bandwidth.
hpGatewayBandwidthTotal	Sum of gateway bandwidth.

## Configuration Tree

The hpConfig tree contains objects that map to Helix Proxy configuration variables. You can monitor these objects using any version of SNMP.

<b>Configuration Tree Objects</b>	
Object	Information
hpMaxConnections	Maximum number of clients that Helix Proxy will accept.
hpMaxGatewayBandwidth	Maximum gateway bandwidth allowed by Helix Proxy.
hpMaxProxyBandwidth	Maximum Helix Proxy bandwidth allowed.



## APPENDIXES

The following appendixes contain useful reference information.



## CONFIGURATION FILE

When you start Helix Proxy, it reads a configuration file to gather system settings. When you change Helix Proxy configuration information, Helix Administrator updates the configuration file automatically. This appendix provides general information about the configuration file.

**For More Information:** For details about configuration lists and variables, see *Helix Proxy Configuration and Registry Reference*.

### Understanding the Configuration File

The configuration file holds the Helix Proxy information in a series of XML-formatted lists and variables. The default file is `rmproxy.cfg`, but you can specify an alternate file at startup, as described in “Starting Helix Proxy” on page 33. The alternate file might be one that you have manually edited, using `rmproxy.cfg` as a starting point.

The Helix Proxy installation directory contains a backup copy of the configuration file named `default.cfg`. This is a mirror image of the default `rmproxy.cfg` file that was created during installation. You can restore your configuration file from the backup if you make changes that you want to undo, or if you accidentally delete the main copy.

**Note:** Be sure to store the configuration file where only authorized users can make changes to it. The default location is the main Helix Proxy’s installation directory.

**Tip:** If you have multiple servers, you may want to name each configuration file differently to identify which server you’re working with.

## Editing the Configuration File

You can change Helix Proxy settings by editing the configuration file with any text editor or XML editor. Some third-party plug-ins may require that you add parameters and variables manually to the configuration file, for example. The configuration file's tags are based on XML (eXtensible Markup Language), and the file is organized into sections for clarity. There are four types of tags in the file:

1. XML declaration tag
2. optional comment tags
3. list tags
4. variable tags

Of these four types, only lists and variables make up the instructions to Helix Proxy. All values for lists and variables are enclosed in double quotation marks.

**Tip:** When you edit the configuration file manually, be sure to use correct syntax. Helix Proxy looks for exact spellings and correct use of angle brackets. Helix Proxy does not display messages related to syntax errors. Instead, it ignores any settings that it does not recognize.

**Note:** Because Helix Administrator reflects the settings of the configuration file in use, exit Helix Administrator before opening the configuration file with a text editor.

### XML Declaration Tag

The XML declaration tag indicates which version of XML is in use. Helix Proxy uses XML version 1.0. The declaration tag looks like this:

```
<?XML Version="1.0" ?>
```

### Comment Tags

Optional comment tags are used in the configuration file to identify tag functions. Identical to comment tags in HTML, they begin with `<!--` and end with `-->`. For example, the following comment tag lets the administrator know that the parameters after it refer to the path settings:

```
<!-- P A T H S -->
```

To disable a feature, convert the feature's tag or tags to a comment. Rather than converting each tag to a comment, you can place the comment's begin tag in front of the feature's first opening tag, and the comment's end tag after the feature's closing tag:

```
<!-- The following feature is commented out
...feature lists and tags here...
-->
```

**Warning!** Do not nest comment tags within other comment tags.

## List Tags

Lists are used for instructions that have several parts, such as the MIME types or the multicast instructions. A list tag is followed by one or more list tags or variable tags. The list tag uses the following syntax:

```
<List Name="name">
...
</List>
```

Here, *name* is the list title. Using the correct capitalization for *name* is important.

Other lists or variables follow the list. The `</List>` tag signifies the end of the list. If other lists are inside the original list, they must also have closing `</List>` tags. The `MIMETypes` list is an example of a list that contains other lists.

**Tip:** Indenting list items is not required, but is recommended for clarity.

## Variable Tags

Variable tags use the following syntax:

```
<Var name="value"/>
```

Here, *name* is the variable name, and *value* is a string or a number, depending on the variable. Capitalization for both *name* and *value* is important. Unlike lists, variables do not have a closing tag; instead, a forward slash mark (/) appears before the closing angle bracket (>).

Variables can be independent elements (such as `LogPath`), or they may appear inside a list. When variables appear within a list, their meaning is determined

by the value of the list name, even though they may appear identical in syntax to variables that are not inside lists. If there are multiple variables within a list that do similar things, their names must be unique. For example, the Extension variables within each MIMETypes list must have different names. This is accomplished by adding a number to the end of each, such as Extension\_01, Extension\_02, and so on.

**Tip:** If you've restarted Helix Proxy and it's not responding to a change you've made to a variable, make sure the variable has a closing forward slash mark, and that there is no space between them.

## Helix Proxy Restart

You typically need to restart Helix Proxy, as described in “Restarting Helix Proxy” on page 41, after you modify the configuration file manually. If you change the Helix Proxy file manually on a UNIX computer, you can use SIGHUP to upload the changes to Helix Proxy without breaking any open connections, as long as the changes do not require a full server restart.

To have Helix Proxy re-read the configuration file, use the following SIGHUP command:

```
kill -HUP processID
```

in which *processID* is the Helix Proxy process number, as shown in the Logs/rmproxy.pid file. For more on this, see “Process ID (PID)” on page 36.

**Note:** When you issue the SIGHUP command, Helix Proxy closes current log files and opens the log files specified in the updated configuration. If log file names have changed, Helix Proxy creates the new logs under the new names.

**Tip:** Mount point changes typically require a full restart. Helix Administrator indicates when configuration changes require a full restart. Use it as your guide to changes that you can and cannot upload with SIGHUP.

## ADDRESS SPACE BIT MASKS

In the multicasting and access control features of Helix Proxy, you can identify a range of IP addresses by assigning a bit mask to an IP address. Helix Proxy interprets the bit mask as a single, contiguous block of address spaces. This appendix describes how to create a bit mask for the purpose of identifying a range of IP addresses.

### Understanding Basic IP Address Construction

To understand how bit masks work, it is helpful to review the basic concepts for constructing an IP address. Each IP address is 32 bits, divided into four 8-bit octets. Each bit in an octet is assigned a value between 128 and 1, from left to right. To indicate whether a value is in use, the bit is set to 1. The sum of all bit values for each octet determines the octet's dotted decimal value. The following defines values for each bit in an octet:

	<b>Bit 1</b>	<b>Bit 2</b>	<b>Bit 3</b>	<b>Bit 4</b>	<b>Bit 5</b>	<b>Bit 6</b>	<b>Bit 7</b>	<b>Bit 8</b>
<b>Bit Value:</b>	128	64	32	16	8	4	2	1

It is possible to make any number between 0 and 255 simply by indicating whether each bit in an octet is set to 1 or 0. For example, both of the following expressions indicate the same IP address:

dotted decimal:	192.0.1.2
32-bit binary equivalent:	11000000 00000000 00000001 00000010

### Using a Bit Mask to Identify an Address Space

To indicate a range of IP addresses, you must first identify the *lowest* IP address in your range, and then indicate the number of bits that are identical between that address and the highest IP address in the range. Consider the range of IP addresses 192.0.1.255 to 192.0.1.0.

These two addresses indicate a range of 256 possible address (in practice only 254, because the all-zero and all-one addresses are reserved). Between the two indicated addresses, 192.0.1.0 is the lowest in the range, and the first three octets (the first 24 bits) are exactly the same for both addresses. Consider the following addresses and the bit mask expressed in binary.

**Addresses and Bit Mask**

Address and Mask	Dotted Decimal	32-Bit Binary Equivalent
Highest Address	192.0.1.255	11000000 00000000 00000001 11111111
Lowest Address	192.0.1.0	11000000 00000000 00000001 00000000
Bit Mask	24 Bits	11111111 11111111 11111111 00000000

Notice that the first 24 bits in the highest and lowest addresses are exactly the same. The same would be true if you had used an address with any decimal number (0-255) in the last octet. The bit mask uses 1's to indicate bits to be evaluated, and 0's to indicate bits to be masked. Thus, assigning a bit mask of 255.255.255.0 to the lowest IP address in the range indicates an address space of 256 possible IP addresses.

## Slash Notation

In the preceding table, the bit mask appears in both its dotted decimal and 32-Bit binary form. However, this same address space can also be articulated with *slash notation* like this:

192.0.1.0/24

Slash notation uses the lowest IP address in the range, followed by a slash and a number that indicates how many bits should be evaluated. This is helpful to understand because in Helix Proxy you indicate a bit mask in a similar manner. You select the number of bits—from 0 bits through 32 bits—from a pull-down list.

## Address Space Size

The size of the address space is determined by the number of bits included in the bit mask. The fewer bits used, the more addresses that are included in the address space. An 8-bit mask includes  $2^{24}$  power addresses, while a 24-bit mask includes only  $2^8$  power addresses.

## Bit Boundaries

Bit boundaries also affect which address can be included in an address space. To understand how bit boundaries work, recall that each octet includes 8 bits, and that each bit has an assigned value. Ranges correspond to the value of each bit in an octet. Further, these ranges cannot cross bit boundaries. Consider the following addresses:

### Bit Boundary with an Inappropriate Mask

Address and Mask	Dotted Decimal	32-Bit Binary Equivalent
Highest Address	192.0.0.2	11000000 00000000 00000000 00000010
Lowest Address	192.0.0.1	11000000 00000000 00000000 00000001
Bit Mask	31 Bits	11111111 11111111 11111111 11111110

Although there are only two consecutive addresses in the range shown in the preceding table, you cannot create a range of two with these addresses, because bit 31 is different for each address. This is a bit boundary. To create a range for these addresses, use the sixth bit in the fourth octet, or a bit mask of 30. Note, though, that by using 30 bits, you also end up including more addresses:

### Bit Boundary with an Appropriate Mask

Address and Mask	Dotted Decimal	32-Bit Binary Equivalent
Highest Address	192.0.1.3	11000000 00000000 00000001 00000011
	192.0.1.2	11000000 00000000 00000001 00000010
	192.0.1.1	11000000 00000000 00000001 00000001
Lowest Address	192.0.1.0	11000000 00000000 00000001 00000000
Bit Mask	30 Bits	11111111 11111111 11111111 11111100

## Determining Bit Boundaries

The chart below identifies every literal bit range available. Look up the bit for the octet you are working with in the **Bit** column on the left. Then use the corresponding **Literal Bit Range** column to look up the decimal values available for each range.

For example, the problem described above arose from attempting to use bit 7 in the fourth octet (Bit 31). However, in row 7 of the table below, no range includes decimal 1 through 2. For a range that works, you need to use the

sixth bit in the fourth octet (Bit 30). Notice that in row 6, there is a decimal range that includes 1 through 2 (range 0-3).

#### Literal Bit Ranges

Bit	Literal Bit Ranges
1	Ranges of 0-127; or 128-255
2	Ranges of 0-63; 64-127; 128-191; or 192-255
3	Ranges of 0-31; 32-63; 64-95; 96-127; 128-159; 160-191; 192-223; 224-255
4	Ranges of 0-15; 16-31; 32-47; 48-63; 64-79; 80-95; 96-111; 112-127; 128-143; 144-159; 160-175; 176-191; 192-207; 208-223; 224-239; 240-255
5	Ranges of 0-7; 8-15; 16-23; 24-31; 32-39; 40-47; 48-55; 56-63; 64-71; 72-79; 80-87; 88-95; 96-103; 104-111; 112-119; 120-127; 128-135; 136-143; 144-151; 152-159; 160-167; 168-175; 176-183; 184-191; 192-199; 200-207; 208-215; 216-223; 224-231; 232-239; 240-247; 248-255
6	Ranges of 0-3; 4-7; 8-11; 12-15; 16-19; 20-23; 24-27; 28-31; 32-35; 36-39; 40-43; 44-47; 48-51; 52-55; 56-59; 60-63; 64-67; 68-71; 72-75; 76-79; 80-83; 84-87; 88-91; 92-95; 96-99; 100-103; 104-107; 108-111; 112-115; 116-119; 120-123; 124-127; 128-131; 132-135; 136-139; 140-143; 144-147; 148-151; 152-155; 156-159; 160-163; 164-167; 168-171; 172-175; 176-179; 180-183; 184-187; 188-191; 192-195; 196-199; 200-203; 204-207; 208-211; 212-215; 216-219; 220-223; 224-227; 228-231; 232-235; 236-239; 240-243; 244-247; 248-251; 252-255
7	Ranges of 0-1; 2-3; 4-5; 6-7; 8-9; 10-11; 12-13; 14-15; 16-17; 18-19; 20-21; 22-23; 24-25; 26-27; 28-29; 30-31; 32-33; 34-35; 36-37; 38-39; 40-41; 42-43; 44-45; 46-47; 48-49; 50-51; 52-53; 54-55; 56-57; 58-59; 60-61; 62-63; 64-65; 66-67; 68-69; 70-71; 72-73; 74-75; 76-77; 78-79; 80-81; 82-83; 84-85; 86-87; 88-89; 90-91; 92-93; 94-95; 96-97; 98-99; 100-101; 102-103; 104-105; 106-107; 108-109; 110-111; 112-113; 114-115; 116-117; 118-119; 120-121; 122-123; 124-125; 126-127; 128-129; 130-131; 132-133; 134-135; 136-137; 138-139; 140-141; 142-143; 144-145; 146-147; 148-149; 150-151; 152-153; 154-155; 156-157; 158-159; 160-161; 162-163; 164-165; 166-167; 168-169; 170-171; 172-173; 174-175; 176-177; 178-179; 180-181; 182-183; 184-185; 186-187; 188-189; 190-191; 192-193; 194-195; 196-197; 198-199; 200-201; 202-203; 204-205; 206-207; 208-209; 210-211; 212-213; 214-215; 216-217; 218-219; 220-221; 222-223; 224-225; 226-227; 228-229; 230-231; 232-233; 234-235; 236-237; 238-239; 240-241; 242-243; 244-245; 246-247; 248-249; 250-251; 252-253; 254-255
8	Only an exact match is possible.

## Working with 0-Bit and 32-Bit Masks

There are two masks that create somewhat special cases: 0-Bits and 32-Bits. When an IP address has a 32-Bit mask, it creates a literal range of 1. For example, consider the following address and 32-Bit mask:

192.0.1.1 /32

When Helix Proxy evaluates incoming IP addresses against this IP address, there is only one possible match: 192.0.1.1. For a match, the incoming address must match all 32-Bits in the original address.

Just as there is only one possible match for addresses with a 32-Bit mask, the opposite is true for addresses with a 0-Bit mask. An IP address with a 0-bit mask essentially tells Helix Proxy to match any addresses. Although not required, you should also enter an origin address of all zeros, like so:

0.0.0.0 /0

This works because Helix Proxy uses a Boolean *and* operation to evaluate incoming addresses. In this type of algorithm, anything and zero equals zero, so all incoming addresses end up equal to the all-zero address entered as the origin address.



## AUTHENTICATION DATA STORAGE

This chapter describes the data storage methods that you can use with the authentication feature described in Chapter 11.

### Understanding Authentication Data

To authenticate visitors, Helix Proxy stores user IDs and passwords. When a client makes a request for media, Helix Proxy looks up this information to see whether the client or visitor is authorized. The information can be stored in either a series of text files or in a database. Templates for common databases are installed during installation:

- **Text file storage**—This default method uses a combination of directory structure and text files to achieve a sensible data storage method. See “Using Text Files for Authentication Data” on page 229 for details.
- **Database templates**—the supplied templates use a similar structure to the text file method, in more familiar database formats. Refer to “Using a Database for Authentication Data” on page 232 for more information.

**For More Information:** For background on using databases with the authentication feature, see “Using Databases” on page 147.

### Using Text Files for Authentication Data

The default configuration uses the text file storage method to provide storage for both default realms. The directories described in the following table

contain the text files which store data. The center letter indicates the authentication protocol: r is for RN5, b is for Basic.

#### Supplied Data Storage Directories

Directory Name	Data Storage for the following type of information
adm_b_db	Helix Administrator user authentication
con_r_db	connection authentication

The following table describes the contents of these directories.

#### Text File Storage Directory Structure

Directory	Contents	File or Directory Description
Main directory (con_r_db or adm_b_db)	ppvbasic.txt	The text file indicates to Helix Proxy that this is the storage area for the list of authenticated names.
users	(initially blank)	Files in this directory list the clips and permission types.
logs	access.txt	See “Logs Directory” on page 231 for a description.
guid	(initially blank)	For player validation, files contain GUIDS to identify individual players.
redirect	(initially blank)	For player validation, files contain a URL to which to send the client if redirection is necessary.

When Helix Proxy creates the file structure, it creates the ppvbasic.txt file. The second and subsequent times you start Helix Proxy, the program looks for this file. If the file does not exist, it recreates the directory structure.

**Warning!** Do not delete the ppvbasic.txt file! If you delete the ppvbasic.txt file, Helix Proxy will rewrite the directories and erase their prior content.

## Users Directory

The files in this directory are named *username*, where *username* is the user’s supplied name. This directory contains one file per registered user. The first line of each file has the following format:

```
password;uuid;uuid_writeable
```

This line contains the following variables:

<i>password</i>	When user authentication is in use, this stores the password. Otherwise shows an asterisk (*). Passwords are encrypted. To change them manually, refer to “Using the Password Tool” on page 146.
<i>uuid</i>	In player validation, this field stores the player ID. In user authentication, an asterisk (*) appears in this field.
<i>uuid_writeable</i>	A flag set and used by Helix Proxy. A value of 0 means that playerID is in the database. A value of 1 means that the record is created, but playerID is not yet registered.

**Note:** If you manually edit the files, be sure that any blank (or unused) fields use an asterisk (\*) as a placeholder. Do not use a space for a placeholder.

## Logs Directory

This directory contains `access.txt`, which is not created until authentication is enabled and the first user connects to Helix Proxy. Each line of `access.txt` describes the result of an attempt to view a clip. The following is the syntax for this file:

```
status;userid;uuid;ip;url;access_type;permission_on;start_time;end_time;total_time;
why_disconnect
```

Each line has the following variables:

<i>status</i>	Result of user’s attempt to connect. If 0, access to the clip is granted. If 1, access is denied
<i>userid</i>	Unique name of up to 50 characters.
<i>uuid</i>	playerID
<i>ip</i>	IP address from which user is attempting to connect.
<i>url</i>	Secured clip user is attempted to access.
<i>permission_type</i>	Event value.
<i>permission_on</i>	Always 0.
<i>start_time</i>	Time/date clip started playing.
<i>end_time</i>	Time/date clip stopped playing.
<i>total_time</i>	Total time clip played.
<i>why_disconnect</i>	Reasons for disconnection. A value of 0 means the client disconnected voluntarily. A value of 1 means that server access expired.

## Using a Database for Authentication Data

This section describes the structure of the ODBC database templates included with Helix Proxy.

To set up the database on Windows and UNIX, see “Setting Up Other Types of Data Storage” on page 233.

The database templates include these tables:

- **Users table**—Lists who is registered and with what access.
- **Access\_log table**—Used by this feature.

### Users Table

This table lists user names and passwords.

Users Table	
Field	Description
<i>userid</i>	User name of up to 50 characters. Ties to permissions table.
<i>password</i>	In user authentication, this stores the password. Otherwise blank. Passwords are encrypted. To change them manually, refer to “Using the Password Tool” on page 146.
<i>uuid</i>	In player validation, stores clientID. In user authentication, an asterisk (*) appears in this field.
<i>uuid_writeable</i>	A flag set and used by Helix Proxy: A value of 0 means that clientID is in the database. A value of 1 means that the record has been created but the clientID is not yet registered with Helix Proxy.

### Access\_log Table

This table shows which restricted sites have been accessed.

Access_log Table	
Field	Description
<i>status</i>	Result of user’s attempt to connect. A value of 0 means that access to the clip was granted. A value of 1 means that it was denied.
<i>userid</i>	Unique name of up to 50 characters.
<i>uuid</i>	Stores player ID.

(Table Page 1 of 2)

**Access\_log Table (continued)**

Field	Description
<i>ip</i>	IP address from which the user is attempting to connect.
<i>url</i>	Secured clip user is attempted to access.
<i>permission_type</i>	Event value.
<i>permission_on</i>	This field is always 0.
<i>start_time</i>	Time/date clip started playing.
<i>end_time</i>	Time/date clip stopped playing.
<i>total_time</i>	Total time clip played.
<i>why_disconnect</i>	Reason for disconnection. A value of 0 means that the client disconnected voluntarily. A value of 1 means that the server access expired.

(Table Page 2 of 2)

## Setting Up Other Types of Data Storage

The following sections explain how to set up alternative data storage methods.

### ODBC Databases

- To set up your Windows computer for ODBC compliance:
  1. On the **Start** menu, point to **Settings**, and click **Control Panel**.
  2. Under **Administrative Tools**, double-click **Data Source (ODBC)**.
  3. On the **System DSN** tab, click **Add**.
  4. Select your ODBC driver from the list of drivers and click **Finish**.
  5. In the **ODBC SQL Server Setup** dialog box, type the data source name. Click **Select**.
  6. Type or browse for the path to your database file and click **OK**.
  7. Click **OK** to exit the ODBC Data Source Administrator.

**Note:** You must now tell Helix Proxy where to find your database. Refer to “Using Databases” on page 147.

## mSQL

Helix Proxy includes a file that sets up the mSQL tables. You can download the mSQL database from the following site:

**<http://www.hughes.com.au/products/msql/>**

► To set up the mSQL database on UNIX:

1. At a command line, start the database application by typing the following:

```
./msql2d &
```

2. Create a database by typing the following:

```
./msqladmin create database_name
```

**Note:** The name you use for *database\_name* must match the database name configured in Helix Administrator.

3. Copy the text file `ppvdemo.db` to your working directory. This file is located in the `./Commerce/database/msql/` subdirectory of the Helix Proxy installation directory.

4. Type the following to set up the table structure:

```
./msql -h localhost database_name < ppvdemo.db
```

**Note:** Be sure to include the less-than sign (`<`).

## GLOSSARY

### **A**    **access control**

A Helix Proxy feature that allows or denies connections based on the requesting client's IP address.

### **ASX file**

A text file that uses the file extension .asx. It launches Windows Media Player and gives it the URL to a streaming clip or presentation.

### **authentication**

A Helix Proxy feature that allows or denies media requests based on the viewer's user name (or global ID) and password.

### **B**    **back-channel**

A control connection to Helix Proxy that a media player maintains during a multicast. The connection allows the player to send commands such as **Stop**, as well as report quality of service.

### **bandwidth**

The upper limit on the amount of data, typically expressed as Kilobits per second (Kbps), that can pass through a network connection.

### **binary tag**

An XML tag that comprises opening and closing tags, such as <List> and </List>.

### **bit**

The smallest unit of measure of data in a computer. A bit has a binary value, either 0 or 1.

### **bit mask**

A code that indicates a range of IP addresses.

### **bit rate**

A measure of bandwidth, expressed as the number of bits transmitted per second. A 28.8 Kbps modem, for example, can transmit or receive around 29,000 bits per second.

**broadcast**

To deliver a presentation, whether live or prerecorded, in which all viewers join the presentation in progress. Broadcast streams can be delivered by unicast or multicast. Contrast to *on-demand*.

**buffering**

The receiving and storing of data before it is played back. A clip's initial buffering is called *preroll*. After this preroll, excessive buffering may stall the presentation.

**byte**

A common measurement of data. One byte consists of 8 bits.

**C**

**cable modems**

Devices that allow rapid transmission and reception of data over television cable. They are digital devices, unlike dial-up modems, which transmit analog data.

**cache**

1. To store a local copy of a clip that resides on a different server.
2. The pool of stored, local clips.

**CBR**

Constant Bit Rate. A type of RealVideo encoding in which all parts of the video play back at the same bit rate. Contrast to *VBR*.

**client**

A software application that receives data from a server. A Web browser is a client of a Web server. RealPlayer is a client of Helix Proxy.

**clip**

A media file within a presentation. Clips typically have an internal timeline, as with RealAudio and RealVideo. Other clip types, such as RealText and SMIL, have a timeline set through markup.

**codec**

Coder/decoder. Codecs convert data between uncompressed and compressed formats, reducing the bandwidth a clip consumes. All audio and video files encoded in a streaming format are compressed using a codec.

**commerce rule**

Rules that determine whether authentication is required for certain on-demand clips or live broadcasts.

- D**    **download**  
To send a file over a network with a nonstreaming protocol such as HTTP. Contrast to *stream*.
- DSL**  
Digital Subscriber Line. A technology for transmitting digital data over a regular telephone line much faster than through dial-up modems.
- duress stream**  
A low-bandwidth audio or video stream that Helix Proxy uses if a connection's available bandwidth drops greatly.
- E**    **encoder**  
Software that generates a live or simulated live stream. For RealMedia, an encoder can be RealProducer or the simulated live transfer agent (SLTA).
- encoding**  
Converting a file into a compressed, streaming format. For example, you can encode .wav files as RealAudio clips.
- F**    **Flash**  
A software application and an animation format created by Macromedia. RealPlayer can play Flash animations and stream them in parallel with other clips, such as RealAudio clips.
- firewall**  
A hardware device or software program that monitors and controls connections between computers and the Internet.
- Flash Player file**  
A compressed Flash file format (file extension .swf) suitable for streaming. To stream Flash, you export the Flash Player file and tune it so that it plays well in RealPlayer.
- H**    **Helix Administrator**  
The browser-based application that you use to configure and run Helix Proxy.
- Helix Proxy**  
RealNetworks software used to stream multimedia presentations to media players. The proxy re-serves streams the originate from Helix Server.

**Helix Server**

RealNetworks server software used to stream multimedia presentations to media players by way of Helix Proxy.

**HTTP**

Hypertext Transport Protocol. The protocol used by Web servers to communicate with Web browsers. In contrast, Helix Proxy streams clips to RealPlayer with RTSP.

**I IETF**

Internet Engineering Task Force. A standards body that proposes and ratifies Internet standards, including protocols such as RTSP and RTP. The IETF maintains a Web site at <http://www.ietf.org/>.

**IP address**

An address expressed in dotted decimal form (as in 123.45.123.45) that identifies a computer on a TCP/IP network.

**ISDN**

Integrated Services Digital Network. Technology that makes digital data connections at 64 or 112 Kbps possible over telephone lines.

**K kilobit (Kb)**

A common unit of data measurement equal to 1024 bits. A kilobit is usually referred to in the context of bit rate per unit of time, such as Kilobits per second (Kbps).

**kilobyte (KB)**

A common unit of data measurement equal to 1024 bytes or 8 Kilobits.

**L LAN**

Local Area Network. A computer network confined to a local area, such as a single building. LANs vary in speed, with bandwidth shared among all networked devices.

**lossy**

A compression scheme that lowers clip size by discarding nonessential data from the source file. Both RealAudio and RealVideo are lossy.

**M MMS**

A proprietary control protocol used for streaming Windows Media clips and broadcasts to Windows Media Player.

**MPEG**

A set of standards-based audio and video compression schemes that includes MP3 and MPEG-4.

**multicast**

Delivering a broadcast so that all media players connect to a single stream instead of receiving a separate stream from the server. Contrast to *unicast*.

**O on-demand**

A type of streaming in which a clip plays from start to finish when a user clicks a link. Most clips are streamed this way. Contrast to *broadcast*.

**P permissions**

Authorizations within the authentication feature that attach to commerce rules to govern which users can view which protected clips or broadcasts.

**PNA**

A discontinued, proprietary protocol Helix Proxy formerly used for backward compatibility with RealPlayer 3 through 5.

**port**

A connection to a server, designated by a number such as 8080. Helix Proxy uses different ports for the RTSP, HTTP, MMS, and PNA protocols.

**preroll**

Buffering that occurs just before a clip plays back. Preroll should be no more than 15 seconds.

**proxy**

A software component that contacts Helix Server to fulfill media requests for media players located behind a firewall.

**Q QuickTime**

A video file format developed by Apple Computer, Inc. and streamed by Helix Proxy. A QuickTime clip can use a variety of codecs, such as the proprietary Sorenson codec or the standards-based MP3 codec.

**R Ram file**

A text file that uses the file extension `.ram` or `.rpm`. It launches RealPlayer and gives it the URL to a streaming clip or presentation.

**RDF**

Resource Description Framework. A mechanism that media players can use to describe their streaming capabilities to Helix Proxy. An XML-based RDF file uses the file extension `.rdf`.

**RDT**

RealNetworks Data Transport. The proprietary data package Helix Proxy uses (along with RTSP) when communicating with RealPlayer. Contrast to *RTP*.

**RealAudio**

A clip type for streaming audio over a network. RealAudio clips use the `.rm` extension.

**realm**

Used with the authentication feature, the realm indicates the database that stores a user's name and password.

**RealPix**

A clip type (file extension `.rp`) for streaming still images over a network. RealPix uses a markup language for creating special effects such as fades and zooms.

**RealPlayer**

The RealNetworks desktop media player that combines streaming and digital download technologies.

**RealProducer**

The primary RealNetworks tool for encoding RealAudio and RealVideo clips.

**RealText**

A clip type (file extension `.rt`) for streaming text over a network. It uses a markup language for formatting text.

**real-time**

Delivered as it occurs. For example, a live event is streamed across a network in a real-time broadcast.

**RealVideo**

A clip type for streaming video over a network. RealVideo clips use the extension `.rm`.

**rebuffering**

An undesirable state in which a media player must pause a presentation to wait for streaming data to arrive. Rebuffering can result from network conditions, or a poorly produced presentation.

**RTCP**

Real-Time Control Protocol. A control protocol used for monitoring and control of RTP sessions.

**RTP**

Real-Time Transport Protocol. The open, standards-based data package Helix Proxy uses (along with RTSP) to communicate with RTP-based clients. Contrast to *RD*T.

**RTSP**

Real-Time Streaming Protocol. An open, standards-based control protocol that Helix Proxy uses to stream clips to RealPlayer or any RTP-based client. Contrast to *HTTP*.

**S SDP file**

A text file that uses the file extension *.sdp*. It provides media players (typically those playing the MPEG format) with information about the clip or broadcast.

**server**

1. A software application, such as a Web server or Helix Proxy, that sends requested data over a network.
2. A computer that runs server software.

**Shockwave Flash**

See *Flash Player file*.

**SMIL**

Synchronized Multimedia Integration Language. A markup language for specifying how and when each clip plays within a presentation. SMIL files use the extension *.smil*.

**stream**

1. To send a media clip over a network so that it begins playing back as quickly as possible.
2. A flow of a single type of data, measured in Kilobits per second (Kbps). A RealVideo clip's soundtrack is one stream, for example.

**SureStream**

A RealNetworks technology that enables a RealAudio or RealVideo clip to stream at multiple bit rates.

**T TCP**

Transmission Control Protocol. An Internet transport protocol that provides a bi-directional channel, allowing Helix Proxy and media players to communicate with each other. Contrast to *UDP*.

**TurboPlay**

A RealPlayer feature that minimizes the amount of initial buffering when a clip begins to play.

**U UDP**

User Datagram Protocol. An Internet transport protocol that allows Helix Proxy to send data to media players more efficiently than when using TCP.

**unary tag**

An XML tag that includes a closing slash, as in `<Var.../>`.

**unicast**

Delivering a separate broadcast stream to each media player. This is the default broadcasting method. Contrast to *multicast*.

**URL**

Uniform Resource Locator. A location description that enables a Web browser or RealPlayer to receive a clip stored on a Web server or Helix Server.

**V VBR**

Variable Bit Rate. A type of RealVideo encoding that enables RealPlayer to play different parts of the video at different bit rates, even though the video is streamed at a constant rate. Contrast to *CBR*.

**W W3C**

World Wide Web Consortium. A standards body that proposes and ratifies Internet software standards, including markup languages such as SMIL. The W3C maintains a Web site at <http://www.w3.org/>.

**Windows Media**

A proprietary audio and video format developed by Microsoft, Inc. Helix Proxy streams the Windows Media format to Windows Media Player.

**X XML**

Extensible Markup Language. The Helix Proxy configuration file is based on XML, which allows one to develop flexible, standardized languages for any purpose.



# INDEX

- A** access control
  - defining rules, 137
  - described, 133
  - Helix Administrator access, 136
  - IPv4 and IPv6 rules, 135
  - predefined rules, 134
  - rule order, 135
- access log
  - accesslog.txt, 231
  - bit rate adaptations, 167
  - client IDs, 164
  - client statistics
    - changes in version 11, 11
    - default value, 180
    - options, 171
    - statistics 1, 172
    - statistics 2, 173
    - statistics 3, 174
    - statistics 4, 175
    - user override, 172
  - customizing, 179
  - described, 155
  - directory, 231
  - firewalls, 128
  - GET statements, 169
  - information fields, 161
  - logging fields
    - final switch, 168
    - session ID, 168
    - switch count, 168
  - logging style, 158
    - style 0, 158
    - style 1, 159
    - style 2, 159
    - style 3, 159
    - style 4, 160
    - style 5, 160
    - style 6, 160
    - style 7, 161
  - media format adaptations, 168
  - presentation ID, 167
  - rolling
    - frequency, 180
    - overview, 157
    - size, 180
- access.txt, 231
- access\_log table, 232
- address space bit masks, 223
- address, *see* IP addresses
- Admin port, 46
- administering Helix Universal Proxy, *see* Helix Administrator
- advanced logging
  - outputs
    - assigning, 191
    - file, 187
    - HTTP post, 188
    - NT event log, 190
    - standard error, 187
    - standard output, 187
    - syslog, 189
    - tcp
      - inbound, 189
      - outbound, 189
    - UDP
      - multicast, 189
      - outbound, 189
  - overview, 183
  - registry, 183
  - templates
    - boilerplate text, 185
    - client stats, 185, 194
    - creating, 190
    - disabling, 191

- formatting, 193
- interval, 184
- new lines, 193
- overview, 184
- proxy configuration changes, 194
- server stats, 193
- session, 186
  - output format, 187
  - watch type, 186
- tabs, 193
- watch, 185
  - output intervals, 191
- variables
  - list of, 192
  - overview, 184
- alternate proxies, *see* redundant proxies
- application-level proxy firewall, 126
- authentication
  - access log, 170
  - databases
    - backing up, 33
    - defining, 148
  - Helix Administrator, 140
  - media players supported, 139
  - on-demand clips, 140
  - overview, 139
  - passwords
    - adding, 143
    - case-sensitivity, 144
    - changing
      - command-line tool, 146
      - Helix Administrator, 145
  - protocols
    - Basic, 150
    - RealSystem 5.0, 150
    - Windows NT LAN manager, 150
  - proxy routing, 99
  - realms
    - creating, 151
    - default realms, 149
    - ID, 151
    - overview, 141
    - protocols, 149
  - user names
    - adding, 143
    - case-sensitivity, 144

- deleting, 144
- listing all, 145
- multiple words, 144
- see also* databases
- automatic bandwidth detection, 23

- B**
- back-channel multicasting
    - access log, 170
    - access rules for player IP addresses, 109
    - broadcast set-up, 108
    - described, 103
    - IP address requirements, 107
    - SureStream, 107
    - time to live
      - definition, 106
      - setting, 108
    - transport, 15
    - unicast failover
      - automatic, 105
      - turning off, 108
  - bandwidth
    - automatic detection, 23
    - client connections, 52
    - conservation through caching, 13
    - gateway maximum, 53
    - no conservation, 22
    - proxy maximum, 53
    - server-side rate control, 61
  - BCC, 89
    - see also* rate control
  - bit masks, 223
  - broadcasting
    - see* live delivery
    - see* multicasting
    - see* pull-splitting
    - see* unicast
  - browser support, 37

- C**
- cache
    - bandwidth conservation, 13
    - described, 19
    - directory, 43
    - disabling, 43
    - proxy routing, 98
    - set-up, 43

- size, 43
    - SureStream, 20
  - capabilities exchange, *see* rate control
  - chaining, *see* proxy routing
  - channel switching
    - logging style, 161
  - child proxy, *see* proxy routing
  - client
    - access to origin server, 18
    - configuration, 111
    - maximum number, 52
    - redirection, 113
    - statistics, *see* access log
  - command-line tools
    - makepass, 146
  - configuration file
    - backing up for reinstallation, 33
    - backup, 219
    - case-sensitivity, 220
    - comment tag, 220
    - editing, 220
    - list tag, 221
    - multiple files, 219
    - security, 219
    - syntax, 220
    - variable tag, 221
  - content caching, *see* cache
  - control protocols, 15
- D**
- data packet formats, 15
  - databases
    - adding to authentication, 148
    - data storage overview, 229
    - default databases, 147
    - flat file, 147
    - mSQL
      - overview, 148
    - ODBC-compliant, 148
      - UNIX setup, 234
    - overview, 141
    - Windows NT LAN manager, 150
  - delayed shutdown
    - defining, 50
    - error log reporting, 50
    - new streams during shutdown, 51
    - playback statistics reporting interval, 50
    - player disconnect interval, 50
    - UNIX shutdown methods, 51
    - Windows shutdown methods, 52
  - delivery methods
    - cache mode, 18
    - pass-through mode, 18
    - pull-splitting mode, 18
  - DES encryption for SNMP, 204
  - differentiated services
    - configuration, 58
    - network requirements, 57
    - overview, 56
    - precedence, 57
    - quality of service, 58
- E**
- error log
    - customizing, 180
    - delayed shutdown statistics, 50
    - file name and location, 181
    - format, 156
    - rolling
      - frequency, 181
      - overview, 157
      - size, 181
    - Windows Event Viewer, 181
  - Extensible Markup Language (XML) *see* XML
- F**
- firewalls
    - described, 119
    - proxy installation issues, 29
- G**
- Gateway installation, 30
  - group variable for UNIX, 53
  - GUID logging, 164
- H**
- Helix Administrator, 22
    - access log, 170
    - activity monitor, 199
    - Admin port, 46
    - configuration file, 23
    - delayed server shutdown, 50
    - manual changes on UNIX, 222
    - password creation, 140
    - starting, 37

- user name creation, 140
- Helix Rate Adaptation, *see* rate control
- Helix Universal Proxy
  - benefits, 13
  - installation directory, 32
  - registry, 183
  - starting on UNIX, 35
  - starting on Windows, 34
  - streaming methods, 18
  - supported media formats, 14
  - upgrading, 32
- HTTP
  - port
    - changing, 46
- HTTP cloaking, 15
- I**
  - installation
    - firewall issues, 29
    - machine ID, 30
    - reinstalling the proxy, 32
    - server and proxy on one machine, 30
    - see also* Gateway installation
  - IP addresses, 48
    - IPv6, 16
      - netmasks, 16
      - shortened addresses, 16
      - unsupported features, 16
    - local host, 48
    - logged for client connections, 162
    - setting, 49
  - ir query string parameter, 71
- J**
  - Java Monitor, *see* Proxy Monitor
- L**
  - Layer-4 switch configuration, 113
  - license file
    - invalid files, 42
    - machine ID, 30
    - multiple files, 42
    - viewing, 41
  - live delivery
    - latency reduction, 21
    - pass-through, 18
    - pull-splitting, 18
    - rate control, 63
  - local host address, 48
  - log file, *see* access log
  - Log Path variable, 36
  - logging
    - see* access log
    - see* advanced logging
    - see* error log
- M**
  - machine ID, 30
  - master agent for SNMP, 207
  - master program on UNIX, 214
  - master.cfg file, 207
  - master.exe program on Windows, 213
  - maximum number of clients, 52
  - MD5 for SNMP, 204
  - media cache, *see* cache
  - media players
    - authentication support, 139
    - configuration, 111
    - default ports, 129
    - maximum number, 52
    - redirection, 113
  - media types, 14
  - memory maximum
    - UNIX, 36
    - Windows, 34
  - MIB file for SNMP, 205
  - Microsoft Media Services (MMS)
    - definition, 122
    - port, 46
  - monitoring proxy activity, 199
  - MPEG
    - authentication, 139
    - MPEG-1 support, 10
    - streaming without a hint track, 15
  - .mrc files
    - see also* rate control
  - multicasting
    - IP addresses, 106
      - IPv6, 106
    - multihomed machines, 107
    - network configuration, 105
    - on the Internet, 105
    - packet time to live (TTL), 106

- multihomed machine multicasting, 107
  - multiple proxies, *see* redundant proxies
  - multi-rate container files
    - see also* rate control
- N**
  - NADU support
    - NADU feedback, 74
  - net masks, 223
  - network address translation firewall, 127
  - network interface card (NIC), *see* multi-homed machines
  - network traffic, 23
- O**
  - ODBC compliance, 233
  - on-demand streaming
    - authentication, 140
    - cache, 18
    - pass-through, 18
- P**
  - packet
    - filter firewall, 127
    - format, 15
    - time to live, 106
  - parent proxy, *see* proxy routing
  - pass-through delivery
    - overview, 22
    - proxy routing, 99
  - passwords
    - changing, 145
    - Helix Administrator, 140
    - media viewers, 139
  - Pid Path variable, 36, 37
  - playlist management, 9
  - PNA, 10
  - ports
    - default, 129
    - defaults
      - media players, 129
      - server, 131
    - defining, 45
    - Helix Administrator, 46
    - HTTP
      - for RDF files, 131
      - setting, 46
    - MMS, 46
    - Proxy Monitor, 46
    - RealPlayer, 112
    - RTSP, 46
    - ppvbasic.txt
      - defined, 230
      - warning, 230
    - privacy policy, 164
    - process id, 36
    - protocols, 15
    - proxy administration, *see* Helix Administrator
    - proxy log, *see* access log
    - Proxy Monitor
      - port, 46
      - using, 199
    - proxy routing
      - authentication, 99
      - caching, 98
      - child proxy, 97
      - defining, 100
      - parent proxy, 97
      - pass-through delivery, 99
      - pull-splitting, 98
      - routing rules
        - examples, 100
        - rule order, 102
      - wildcards, 100
    - pull-splitting
      - modifying splitting, 44
      - overview, 21
      - protocol used, 44
      - proxy routing, 98
      - server resends, 45
      - version 8 servers, 45
- Q**
  - QuickTime authentication, 139
- R**
  - rate control
    - 3GPP adaptation, 62
      - enabling, 90
    - basic requirements, 61
    - BCC
      - enabling, 96
      - overview, 89
      - preset values, 95

- specification, 89
- buffer model, 75
  - buffer size, 76
  - minimum preroll setting, 77
  - preroll, 76
- capabilities exchange, 65, 93
- compatibility
  - media formats, 63
  - media players, 62
- configuring, 89
- congestion control, 88
- default media rate, 92
- downshifting, 79
  - client-initiated, 82
  - downshift depth, 80
    - setting, 92
  - downshift target-time depth, 81
    - setting, 92
  - stepwise downshifting, 81
- encoding guidelines, 64
- Helix adaptation, 62
  - enabling, 90
- initial oversend rate, 72
- initial rate (ir) query string parameter, 71
- initial rate selection, 71
- Link-Char header, 82
- live delivery, 63
  - enabling, 91
- max advance, 86
- maximum sending rate, 70, 92
- NADU feedback, 74
- overview, 61
- presets
  - aggressive, 94
  - BCC values, 95
  - conservative, 94
  - moderate, 94
  - selecting, 91
- profiles
  - base profile selection, 91
  - default profile, 68
  - inheritance, 66
  - overview, 65
  - pre-defined profiles, 69
  - user-agent matching
    - exact, 67
    - partial, 67
- RDF files, 65, 93
- receiver reports, 72
  - receiver report frequency, 73
  - report timeout, 74
  - sender report frequency, 74
- slowdown mode, 85
  - exiting, 87
  - max advance, 86
- stream switch delay, 92
- stream.id evaluation, 71
- SureStream comparison, 64
- target time, 78
- TCP, 89
- TFRC
  - overview, 88
  - specification, 88
- types of, 62
- UAS files, 65
- upshifting, 82
  - client-initiated, 85
  - stepwise upshifting, 84
  - upshift depth, 83
    - setting, 92
  - upshift target-time depth, 84
    - setting, 92
- user-agent strings, 67
  - assigning to profiles, 90
- video decode byte rate, 93
- video predecode buffer size, 93
- x-wap-profile, 93
- RDF files, 65
- RDT, 15
- Real Time Streaming Protocol *see* RTSP
- realms, *see* authentication
- RealPlayer
  - bandwidth detection, 23
  - configuring, 111
  - server-side rate control, 70
- redirection of clients, 113
- redundant proxies
  - overview, 54
  - requirements, 55
  - RTSP requirement, 54
  - setting up, 55

- registry, 183
  - reinstallation, 32
    - authentication database backup, 33
    - backing up configuration file, 33
  - reports
    - see* access log
    - see* error log
    - see* advanced logging
  - rmproxy.pid, 36
  - RTCP
    - rate control, 72
  - RTP, 15
  - RTSP, 122
    - connection timeout, 53
    - Link-Char header, 82
    - port, 46
  - RTSP protocol, 15
- S**
- scalable multicasting time to live, 106
  - server access by proxy clients, 18
  - SHA for SNMP, 204
  - shutdown delay, *see* delayed shutdown
  - SIGHUP command, 222
  - SMIL in access log, 167
  - SNMP
    - AgentX protocol and port, 209
    - authentication, 204
    - DES encryption, 204
    - management system, 214
    - master agent
      - address and port, 209
      - configuration, 207
      - security model, 209
      - UNIX startup, 214
      - VACM setup, 210
      - Windows Service installation, 32
      - Windows startup, 213
    - MD5, 204
    - MIB file, 205
    - overview, 203
    - plug-in configuration, 205
    - privacy, 204
    - security, 209
    - SHA, 204
    - traps
      - bandwidth usage, 207
      - CPU utilization, 206
      - defining, 205
      - enabling, 206
      - interval for, 206
      - memory usage, 207
      - player connections, 206
      - server startup, 206
    - trees
      - configuration, 215
      - monitor, 215
      - VACM setup, 210
      - version support, 204
  - SOCKS firewall, 128
  - starting up
    - UNIX, 35
      - memory maximum, 36
    - Windows, 34
      - memory maximum, 34
  - stateful packet filtering firewall, 127
  - stopping Helix Universal Proxy
    - UNIX, 37
    - Windows, 37
  - stopping the server
    - delayed shutdown, 50
  - supported media types, 14
  - SureStream
    - caching, 20
    - multicast address requirement, 107
    - rate control comparison, 64
- T**
- tables
    - access\_log, 232
    - users, 232
  - TCP transport, 15
  - TCP/IP
    - congestion control, 89
  - TFRC, 88
    - see also* rate control
  - 3GPP rate adaptation, *see* rate control
  - time to live (TTL), 106
  - tracking clip activity, 25
  - transparent proxy firewall, 127
  - troubleshooting guide, 5

TurboPlay statistics, 177

- U**
  - UDP transport, 15
  - unicast failover from multicast
    - description, 105
    - turning off, 108
  - UNIX
    - authentication database setup, 234
    - SIGHUP, 222
    - special features, 53
    - stopping Helix Universal Proxy, 37
    - user and group name, 53
  
- W**
  - Web server installation with proxy, 29
  - wildcards in proxy routing rules, 100
  - Windows
    - Event Viewer, 181
    - proxy service
      - automatic configuration, 32
      - starting the proxy, 33
  - Windows Media
    - authentication, 139
    - time to live, 106
  - Windows Media Player
    - activity logged, 164
    - client statistics, 171
    - configuring version 11 and later, 113
    - configuring versions 7 through 10, 112
    - see also* clients
    - see also* MMS
  
- X**
  - XML
    - comment tag, 220
    - configuration file, 23, 220
    - declaration tag, 220