

NetCache™ 5.4

Guide to Caching Protocols and Services

Network Appliance, Inc.
495 East Java Drive
Sunnyvale, CA 94089 USA
Telephone: +1 (408) 822-6000
Fax: +1 (408) 822-4501
Support telephone: +1 (888) 4-NETAPP
Documentation comments: doccomments@netapp.com
Information Web: <http://www.netapp.com>

Part number 210-00369
December 2002

Copyright and trademark information

Copyright information

Copyright © 1994–2002 Network Appliance, Inc. All rights reserved. Printed in the U.S.A.

Portions copyright © 1998–2001 The OpenSSL Project. All rights reserved.

No part of this book covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Portions of this product are derived from the Berkeley Net2 release and the 4.4-Lite-2 release, which are copyrighted and publicly distributed by The Regents of the University of California.

Copyright © 1980–1995 The Regents of the University of California. All rights reserved.

Portions of this product are derived from NetBSD, which is copyrighted by Carnegie Mellon University.

Copyright © 1994, 1995 Carnegie Mellon University. All rights reserved. Author Chris G. Demetriou.

Permission to use, copy, modify, and distribute this software and its documentation is hereby granted, provided that both the copyright notice and its permission notice appear in all copies of the software, derivative works or modified versions, and any portions thereof, and that both notices appear in supporting documentation.

CARNEGIE MELLON ALLOWS FREE USE OF THIS SOFTWARE IN ITS “AS IS” CONDITION. CARNEGIE MELLON DISCLAIMS ANY LIABILITY OF ANY KIND FOR ANY DAMAGES WHATSOEVER RESULTING FROM THE USE OF THIS SOFTWARE.

Software derived from copyrighted material of The Regents of the University of California and Carnegie Mellon University is subject to the following license and disclaimer:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notices, this list of conditions, and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment:

This product includes software developed by the University of California, Berkeley and its contributors.
4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS

INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions of the software were created by Netscape Communications Corp.

The contents of those portions are subject to the Netscape Public License Version 1.0 (the “License”); you may not use those portions except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/NPL/>.

Software distributed under the License is distributed on an “AS IS” basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is Mozilla Communicator client code, released March 31, 1998.

The Initial Developer of the Original Code is Netscape Communications Corp. Portions created by Netscape are Copyright © 1998 Netscape Communications Corp. All rights reserved.

Software derived from copyrighted material of Network Appliance, Inc. is subject to the following license and disclaimer:

Network Appliance reserves the right to change any products described herein at any time, and without notice. Network Appliance assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by Network Appliance. The use and purchase of this product do not convey a license under any patent rights, trademark rights, or any other intellectual property rights of Network Appliance.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

Trademark information

NetApp and the Network Appliance design are registered trademarks of Network Appliance, Inc. in the United States, Canada, and the European Union. Network Appliance is a registered trademark of Network Appliance, Inc. in Monaco and a trademark of Network Appliance, Inc. in the United States and Canada. FAServer is a registered trademark of Network Appliance, Inc. in the United States and the European Union. NetCache is a registered trademark of Network Appliance, Inc. in the European Union and Japan, and a trademark of Network Appliance, Inc. in the United States. SnapCopy is a registered trademark of Network Appliance, Inc. in the European Union and a trademark of Network Appliance, Inc. in the United States. SnapManager is a registered trademark of Network Appliance, Inc. in the United States. WAFL is a registered trademark of Network Appliance, Inc. in the United States, the European Union, and Canada. FilerView and SecureShare are registered trademarks of Network Appliance, Inc. in the United States. Data ONTAP is a trademark of Network Appliance, Inc. in the United States and Canada. Snapshot is a trademark of Network Appliance, Inc. in the United States and the European Union. NetApp—the Network Appliance Company is a registered trademark of Network Appliance, Inc. in the United States and other countries. ApplianceWatch, BareMetal, Center-to-Edge, ContentDirector, DataFabric, MultiStore, NearStore, SecureAdmin, Serving Data by Design, Smart SAN, SnapCache, SnapDrive, SnapMirror, SnapRestore, SnapVault, Vfiler, and Web Filer are trademarks of Network Appliance, Inc. in the United States.

Apple is a registered trademark and QuickTime is a trademark of Apple Computer, Inc. in the United States and/or other countries.

Microsoft is a registered trademark and Windows Media is a trademark of Microsoft Corporation in the United States and/or other countries.

RealAudio, RealNetworks, RealPlayer, RealSystem, RealText, and RealVideo are registered trademarks and RealMedia, RealProxy, and SureStream are trademarks of RealNetworks, Inc. in the United States and/or other countries.

Network Appliance is a licensee of the CompactFlash and CF Logo trademarks.

All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such.

Network Appliance NetCache is certified RealSystem compatible.

Table of Contents

	Preface	.ix
Chapter 1	Planning for Web Cache Service	1
	Section A: Planning for Web cache configuration	2
	Planning for network deployment configuration	3
	Determining the configuration requirements	4
	Additional features to consider implementing	5
	Section B: How NetCache manages Web objects	7
	Which Web objects are cached	8
	How long objects remain in the cache	12
	How often objects are refreshed	16
	How NetCache handles objects resulting from aborted requests.	17
	Section C: General HTTP configuration options	18
	Section D: About FTP support	21
	Overview of NetCache FTP support	22
	FTP object management	23
	Support for FTP connections	25
	Section E: About tunneling HTTP-based traffic	27
	Section F: Web cache performance	29
	Section G: Monitoring a Web cache	34
Chapter 2	Planning for Streaming Media Service	35
	Section A: Planning for streaming media configuration	37
	Planning for network deployment	38
	Determining the streaming media configuration requirements.	39
	Additional features to consider implementing	40
	Applicability of other NetCache features to streaming.	42
	Section B: Streaming media configuration requirements	44

General streaming media requirements.	45
Configuration recommendations	47
Basic firewall requirements for streaming media	51
MMS streaming requirements	53
RealSystem streaming requirements	54
QuickTime streaming requirements	56
Multicast requirements	58
confill requirements.	59
Section C: Customizing your streaming media configuration	61
Reducing redundant fetching	62
Implementing nontransparent redirection for MMS streaming.	65
Using URLs to direct requests to NetCache	67
Prefilling streaming media content	69
Ensuring that client logs are sent to servers	71
Authenticating streaming client requests	74
Planning for client request volume	75
Understanding streaming media connection statistics	76
Section D: About streaming object management	80
Cacheable streaming media content	81
How NetCache handles streaming objects	83
Section E: Troubleshooting the streaming media configuration	85
Section F: Tuning a streaming media cache for performance	90
Section G: Streaming media protocol details	95
MMS streaming media protocol details	96
RealSystem streaming media protocol details	98
QuickTime streaming media protocol details	100
Understanding HTTP for streaming	102
NetCache behavior for MMS multicast streaming	104

Chapter 3	Planning for Web or Streaming Acceleration	107
	Section A: Planning for accelerator configuration	108
	Gathering preconfiguration information	109
	Planning for uninterrupted service	112
	Section B: Overview of accelerator configuration	113
	Acceleration setup procedure.	114
	DNS configuration to support an accelerator.	118
	Appliance Manager options for configuring acceleration	122
	Continuing the accelerator configuration.	126
	Section C: Accelerator configuration scenarios	127
	Scenario: Configuring an accelerator outside the firewall	128
	Scenario: Deploying an appliance as a distributed content site accelerator . .	131
	Scenario: Using multiple accelerators for a single Web or streaming server .	134
	Scenario: Deploying one accelerator for multiple Web or streaming servers .	138
	Scenario: Allowing limited access from another company.	142
 Chapter 4	 Planning for News Cache Service	 145
	Section A: Planning for news cache configuration	146
	Planning for network deployment configuration	147
	Determining the configuration requirements	148
	Additional features to consider implementing	149
	Section B: How NetCache manages news objects	151
	Which news objects NetCache caches	152
	When NetCache checks for new articles	153
	When NetCache checks for canceled articles	155
	Section C: Configuration options for a news cache	157
	About the setup options.	158
	About the logging options	161
	Section D: News cache performance	162
	Section E: Monitoring a news cache	166

NNTP statistics pages. 167

NNTP access log files 169

Section F: News cache configuration scenario. 171

Index 179

Preface

About this guide

This guide provides information to help you plan the configuration of NetCache™ appliances running NetCache 5.4 software. It provides information for the following supported NetCache platforms: C1100, C1105, C1200, C2100, C3100, and C6100.

For information about how to upgrade or service the system hardware, see the *Field Service Guide* for your NetCache appliance model.

Audience

This guide is for system administrators who are familiar with computer networks, Web technology, and HTTP service. This guide assumes that customers planning to use the NetCache appliance as a news cache have a basic understanding of NNTP and news service. This guide also assumes that customers planning to use the NetCache appliance as a streaming media cache have a basic understanding of RTSP, MMS, and streaming media service. This guide does not cover basic system administration topics such as IP addressing, routing, and network topology. It emphasizes the characteristics of the NetCache appliance hardware and the NetCache software.

Command and keyboard conventions

You can enter NetCache appliance commands on either the system console or from any client computer that can access the NetCache appliance through Telnet.

When describing key combinations, this guide uses the hyphen (-) to separate individual keys. For example, “Ctrl-D” means pressing the “Control” and “D” keys simultaneously.

This guide uses the term “type” to mean pressing one or more keys on the keyboard. It uses the term “enter” to mean pressing one or more keys and then Enter. Also, this guide uses the term “Enter” to refer to the key that generates a carriage return, although the key is named “Return” on some keyboards.

This guide uses capitalization and some abbreviations to refer to the keys on the keyboard. (The keys on your keyboard might not be labeled exactly as they are in this guide.)

Formatting conventions

The following table describes typographical conventions used in this guide.

Formatting convention	Type of information
<i>Italic type</i>	Words or characters that require special attention. Placeholders for information you must supply. For example, if the guide says to enter the <code>arp -d hostname</code> command, you enter the characters “arp -d” followed by the actual name of the host. Book titles in cross-references.
Monospaced font	Command and daemon names. Information displayed on the system console or other computer monitors. The contents of files.
Bold monospaced font	Words or characters you type. What you type is always shown in lowercase letters, unless it must be typed in uppercase letters to work properly.

About this chapter	A NetCache appliance can be used as a Web cache to provide content-caching service to clients on the network. This chapter provides information to help users plan their configurations of their NetCache appliances as Web caches.
Before reading further	This chapter assumes that you have read the information about Web caching in the <i>Deployment Guide</i> and have made decisions about how you want to deploy your NetCache appliance as a Web cache on your network.
Sections in this chapter	This chapter provides information about the following topics: ◆ Section A, “Planning for Web cache configuration,” on page 2 ◆ Section B, “How NetCache manages Web objects,” on page 7 ◆ Section C, “General HTTP configuration options,” on page 18 ◆ Section D, “About FTP support,” on page 21 ◆ Section E, “About tunneling HTTP-based traffic,” on page 27 ◆ Section F, “Web cache performance,” on page 29 ◆ Section G, “Monitoring a Web cache,” on page 34
Procedures and option descriptions	See the Appliance Manager online Help for configuration procedures and option descriptions.
Location of configuration options	To access the HTTP options in the Appliance Manager, select: Setup tab > HTTP To access the FTP options in the Appliance Manager, select: Setup tab > FTP

Section A: Planning for Web cache configuration

About this section This section provides information to help you plan how to configure your NetCache appliance as a Web cache.

Topics in this section This section provides information about the following topics:

- ◆ “[Planning for network deployment configuration](#)” on page 3
- ◆ “[Determining the configuration requirements](#)” on page 4
- ◆ “[Additional features to consider implementing](#)” on page 5

Accessing the Appliance Manager and the CLI After you have planned how to configure your NetCache appliance as a Web cache, you can configure the appliance using the Appliance Manager (the NetCache browser-based administration interface) or the NetCache command line interface (CLI). For information about accessing the Appliance Manager, see the *NetCache Quick Setup* document. For information about the NetCache CLI, see the *Command Line Reference*.

Planning for network deployment configuration

Decisions to be made

While you read the *Deployment Guide*, you made decisions about deploying your NetCache appliance as a Web cache on your network. Some of the deployment decisions you made require configuration using the Appliance Manager or the NetCache CLI.

The following table lists deployment decisions that require configuration, and it provides the location of information to help you plan how to configure the features that you want to implement.

If you want to...	See...
Deploy the appliance as a transparent cache with an L4 or L7 switch or a WCCP 2.0-based router	The transparent proxying chapter in the <i>Administration Guide</i>
Deploy the appliance with a firewall or with no firewall	The hierarchy chapter in the <i>Administration Guide</i>
Implement a load balancing strategy across multiple Web caches	The transparent proxying chapter in the <i>Administration Guide</i>
Use IP takeover pairs to enable one appliance to take over services for another if one fails	The takeover chapter in the <i>Administration Guide</i>
Deploy the appliance with one or more caches upstream	The hierarchy chapter in the <i>Administration Guide</i>

Determining the configuration requirements

Decisions to be made

After you have planned for the basic configuration of your NetCache appliance, you should decide how you want to optimize your NetCache appliance to function as a Web cache on your network.

While you read this chapter, you will determine whether you want to do the following:

- ◆ Specify types of content to be proxied and cached
- ◆ Configure the appliance to drop or maintain specified client connections
- ◆ Specify how the appliance will respond to canceled requests and reload requests
- ◆ Specify object refresh rates
- ◆ Control user access to HTTP services
- ◆ Configure the appliance to not provide the client IP address to the Web server
- ◆ Suppress error messages that refer to the presence of a proxy-cache server to clients
- ◆ Configure the appliance to forward, tunnel, or redirect specified types of traffic
- ◆ Configure the appliance to support FTP

Additional features to consider implementing

Optional features available

NetCache includes a number of features that you can use to customize the NetCache appliance for your environment. Use the following table to help you determine which features to implement.

If you want to...	See...
◆ Customize the configuration of network interfaces ◆ Create virtual interfaces	The network interface chapter in the <i>Administration Guide</i>
Configure NetCache to use default and static routes for incoming and outgoing requests	The routing chapter in the <i>Administration Guide</i>
Configure the appliance for host name resolution and for DNS caching	The DNS chapter in the <i>Administration Guide</i>
Configure the appliance to support FTP	This chapter in this guide
Require users to supply a user name and password when using Gopher	The Gopher topic in the Appliance Manager online Help
Restrict administrative access for security purposes	The administrative security chapter in the <i>Security Guide</i>
Require users to authenticate themselves before NetCache services requests	The authentication chapter in the <i>Security Guide</i>
◆ Prevent unauthorized clients from accessing NetCache services ◆ Restrict the access of authorized clients to specified content and NetCache services	The access control chapter in the <i>Security Guide</i>
Establish direct tunneling through the appliance for traffic that the appliance does not handle	The protocol tunneling chapter in the <i>Administration Guide</i>
Configure the SNMP agent on the appliance to use diagnostic and other management services	The SNMP chapter in the <i>Administration Guide</i>
Enable or disable log files	The log files chapter in the <i>Administration Guide</i>

If you want to...	See...
Select and modify log file formats	The <i>Guide to Messages and Log File Entries</i>
◆ Customize error messages ◆ Hide error messages from users	The error handling chapter in the <i>Administration Guide</i>
Control bandwidth use	The bandwidth allocation topic in the Appliance Manager online Help
Configure the ContentDirector software to distribute content from a central source to remote NetCache appliances	The ContentDirector agent topic in the Appliance Manager online Help
Push HTTP content to the NetCache appliance using the <code>confill</code> command	The confill information in “ Prefilling streaming media content ” on page 69 and the <code>confill</code> CLI help
Enable your NetCache appliance to interact with an ICAP server to adapt content to local policies	The <i>ICAP Services Guide</i>
Modify your Autosupport configuration settings	The Autosupport chapter in the <i>Administration Guide</i>

Section B: How NetCache manages Web objects

About this section This section describes the configuration options that relate to object management services.

Topics in this section This section provides information about the following topics:

- ◆ [“Which Web objects are cached”](#) on page 8
- ◆ [“How long objects remain in the cache”](#) on page 12
- ◆ [“How often objects are refreshed”](#) on page 16
- ◆ [“How NetCache handles objects resulting from aborted requests”](#) on page 17

Which Web objects are cached

How NetCache decides which objects are cached

Clients and Web servers use special tags in objects to instruct Web caches on what to cache and what not to cache. By default, the NetCache appliance does not cache an object if any of the conditions in the following table are met.

Condition	Explanation
An object’s HTTP MIME header contains one of the following: ◆ Cache-Control: no-cache ◆ Cache-Control: no-store ◆ Cache-Control: private ◆ Pragma: no-cache	A MIME header, inserted by the Web server into the object, instructs the NetCache appliance not to save a copy of the object in its cache.
An object’s HTTP MIME header contains a <i>Set-Cookie</i> line.	A Set-Cookie line, inserted by the Web server into the object, indicates that a cookie has been used. A cookie is used to maintain user-specific information, allowing for more personalized content. Because the content is usually specific to a user, caching it uses disk space for objects that will never be requested by another user. Exceptions: If the object contains “Cache-Control: public” in its header, or if the object has an extension that matches an item on the Cookie Cheat File Name Extensions list in the Appliance Manager, NetCache caches the object even if it contains a cookie.

You can define other types of objects as noncacheable

Highly dynamic and user-specific content are not good candidates for caching because the information is unlikely to be requested by another user. One type of request you might want to define as noncacheable, for example, is a `cgi-bin` request. Because `cgi-bin` requests are highly dynamic, they are inherently noncacheable.

Another type of content you might want to define as noncacheable is any content on a Web server that is in the same domain as the NetCache appliance. If a Web server is local, the NetCache appliance can simply proxy requests for the Web server without having local objects use cache disk space.

Using the HTTP - Header and Caching Behavior page in the Appliance Manager, you can define explicit caching rules that specify objects as cacheable or noncacheable by date or time requested, Web server IP address, host name or port, client IP address or port, URL, URL prefix, or content type.

Caching static objects in URLs containing cookies

Some Web servers use cookies to prevent Web caches from caching their Web pages. When a Set-Cookie header is included in the Web server response, the header is stored on the client. All subsequent requests for URLs that match the domain and path specified by the Set-Cookie header cause the client to issue a Cookie request header. Including this header makes these Web servers' pages noncacheable, even for static content.

However, even when some of the content of a Web page is user-specific, the content might contain objects that are static, such as graphics. The NetCache appliance enables you to define these objects as cacheable, even when the surrounding content is user-specific and, therefore, noncacheable.

You can instruct the NetCache appliance to cache specific objects, even when these objects have cookies associated with them, by using the Cookie Cheat File Name Extensions option on the HTTP - Header and Caching Behavior page under the Setup tab in the Appliance Manager. The Cookie Cheat File Name Extensions list causes NetCache to strip out the Cookie request headers so that they are not passed on to the origin server, thereby making these objects subject to normal caching rules.

Rules and options to determine cacheable objects

You can configure what the NetCache appliance does and does not cache by creating and editing explicit caching rules through the Explicit Caching Rules tab on the HTTP - Header and Caching Behavior page, and by configuring additional options also on that page.

About explicit caching rules

You can configure NetCache explicit caching rules to determine which objects NetCache will or will not cache. You can specify by date and time requested, Web server host name, server IP address, server port, client IP address, client port, URL, URL-prefix, or content type which objects fetched through HTTP requests will be cached on the appliance.

To view existing explicit caching rules in the Appliance Manager, select Setup tab > HTTP > Header and Caching Behavior. Click the Explicit Caching Rules tab to display the Current Explicit Rules table. This table lists the current explicit caching rules.

Options available to handle objects

In addition to using explicit caching rules, you can configure other options on the Appliance Manager HTTP - Header and Caching Behavior page to specify the objects that the NetCache appliance does and does not cache.

The options in the following table are available.

If you want to...	Use this option...
Specify that the cache proxies Accept-ranges:byte headers from the origin server	Accept-Ranges Header Forwarding
Specify that objects are cached that do not have a Last Modified Time header or an explicit expiration time	4.1-Compatible Caching Enable
Specify that objects are reloaded in accordance with the listed reload rules	Reload Rules
Specify that the Cache-Control: no-cache header is ignored in objects that the client sends	Ignore No-Cache Directive on Client Reload
Strip the set-cookie header from any cached object containing a cookie	Treating Requests with No Cookie Match
Specify how NetCache handles a client request if the client requests a currently cached object containing a cookie but does not include a cookie in its request	Treating Requests with No Cookie Match
Specify source URLs whose fetched objects will be cached on this NetCache appliance even if they contain cookies	Cacheable Cookie by URL

If you want to...	Use this option...
Specify that objects that end in the listed extensions are cacheable, even if the objects contain a cookie	Cookie Cheat File Name Extensions
◆ Specify that the ? and trailing characters are ignored when caching an object ◆ Specify that items entered in the URL Trailing Characters Substring List are ignored when caching an object	◆ Ignore URL Trailing Characters ◆ Ignore URL Trailing Characters Substring List

How long objects remain in the cache

About object Time-To-Live

NetCache keeps an object in its cache based on the object's maximum age, or Time-To-Live (TTL). How the TTL is calculated depends on whether or not the request is accelerated.

For nonaccelerated requests: An object's TTL is determined by the following criteria, in order of precedence:

- ◆ Cache control headers contained in the object itself that specify an explicit expiration time, which NetCache uses as the object's TTL
- ◆ Settings entered in the TTL Patterns option on the HTTP - Header and Caching Behavior page
- ◆ Default TTL that NetCache calculates using its adaptive algorithm

Note that you can configure NetCache to override the TTL specified by an object's cache control headers through settings on the HTTP - Header and Caching Behavior page.

In addition, you can enter If-Modified_Since (IMS) rules in the IMS rules tab on the HTTP - Header and Caching Behavior page that can specify IMS queries that result in an object being refreshed before its TTL is expired.

For accelerated requests: An object's TTL is determined by the following criteria, in order of precedence:

- ◆ Settings entered in the TTL Patterns option on the HTTP - Header and Caching Behavior page
- ◆ Cache control headers contained in the object itself that specify an explicit expiration time, which NetCache uses as the object's TTL
- ◆ Default TTL that NetCache calculates using its adaptive algorithm

When NetCache verifies an object with the Web server

When a user requests an object that is in the cache and that object has expired, that is, it has exceeded its TTL, NetCache verifies the freshness of the object with its originating Web server. NetCache verifies the freshness of the object by sending a request to the Web server inquiring whether the object has been modified. This query is called an *If-Modified-Since*, or IMS, query.

If the response to the IMS query indicates that the object has not been modified, NetCache delivers the object from its cache and recalculates the object TTL. If the object has been modified, NetCache ejects the older object, caches the updated object, and sends it to the user.

If an object cannot be verified because NetCache fails to connect to the Web server, NetCache returns the last-verified object.

How cache control headers affect the object TTL

When a Web server sends an object to the NetCache appliance, it includes in its headers the object's expiration time and last-modification time. If the object contains an explicit expiration time, NetCache uses it regardless of any entries in the TTL Patterns option.

How NetCache calculates a default TTL for an object

If the object has a specified last-modified time, NetCache calculates the default TTL differently than if the object has no last-modified time.

To determine when NetCache calculates the default TTL, see [“About object Time-To-Live”](#) on page 12.

TTL for objects with no last-modified time: If no last-modified time has been specified for the object, NetCache uses the age specified in the Default Maximum Age option on the HTTP - Header and Caching Behavior page as the initial TTL.

TTL for objects with a last-modified time: If a last-modified time has been specified for the object, NetCache sets the object's initial TTL to a value equal to half the length of time between the object's last modification time and the current time.

Using this adaptive algorithm, objects that have not been changed for a long time tend to have very high default TTLs, and objects that change frequently tend to have very low default TTLs. This approach enables NetCache to maximize bandwidth savings by not refetching objects that have not been modified and by not verifying objects that are unlikely to have been modified. At the same time, this approach helps keep cache content current, or fresh, with the state of the object on the originating Web server.

Example: If an object is cached for the first time with a last-modified date of one day before, NetCache sets the object's expiration time to 12 hours. After the object has been in the cache for 12 hours, the next user request for the object causes NetCache to verify the object with the Web server. If the object has not

changed, NetCache recalculates the expiration time based on the original last modification time, 36 hours ago. NetCache sets the new expiration time for the object to 18 hours.

With this process, if the object continues to be unmodified, the object's expiration time continues to increase with each verification by NetCache. However, if NetCache determines that the object has been modified, NetCache resets the TTL to a relatively low value until the object in its cache has been unmodified for a longer period of time.

About the TTL Patterns option

Using the TTL Patterns option on the HTTP - Header and Caching Behavior page in the Appliance Manager, you can modify the TTL that NetCache uses for objects.

TTL pattern rules are invoked if the URL matches the regular expression specified in the TTL Patterns box. Regular expressions are checked in order; the first expression that matches the URL is used to determine the object's TTL. If the URL does not match any pattern, no TTL pattern rule is invoked.

However, if the Web server included an explicit expiration time in the object, NetCache uses the expiration time regardless of any entries in the TTL Patterns option.

About creating TTL pattern rules

You can create TTL pattern rules using a regular expression of object URLs. Use the following syntax to create TTL pattern rules:

regular_expression absolute_TTL <percentage_TTL> <maximum_TTL>

In the preceding syntax example, *regular_expression* is the URL string; *absolute_TTL* is the absolute TTL in minutes; *percentage_TTL* is a percentage of the object's age; and *maximum_TTL* is the maximum number of minutes an object can reside in the cache before reverification. The relationship between these syntax elements is explained in the following sections.

regular_expression: A regular expression of the URL. For example:

`^http://`

The preceding example matches all HTTP objects, that is, all objects with strings that begin with `http://`. For more information about regular expressions, see the chapter on access control in the *Security Guide*.

absolute_TTL: The absolute number of minutes an object can reside in the cache before reverification. The absolute TTL value is used to calculate an object's TTL when no last-modified time is specified for the object or if the *percentage_TTL* is unspecified or set to zero.

For example:

```
^http:// 1440
```

In the preceding example, HTTP objects (strings that begin with http://) are cached for 1,440 minutes.

An absolute TTL of zero causes NetCache to send an IMS query request to the Web server to verify the freshness of the object.

percentage_TTL and maximum_TTL: NetCache uses the *percentage_TTL* variable in conjunction with the *maximum_TTL* variable to calculate an object's TTL as follows:

$$(\text{current time} - \text{last modified time}) \times (\text{percentage_TTL} / 100)$$

NetCache calculates the *percentage_TTL*, compares it to the value specified for the *maximum_TTL*, and uses the lesser of the two values.

Example: Assume that the following rule is entered:

```
^http:// 0 20% 4320
```

If the object is cached at 9:00 a.m. and the object was last modified 7:00 p.m., NetCache calculates 20 percent of 14 hours (9:00 a.m. minus 7:00 p.m.), to arrive at a total of 168 minutes. NetCache then compares the total found in the *percentage_TTL* calculation (168 minutes) to the *maximum_TTL* (4320 minutes) and uses the lesser value. Therefore, NetCache sets the object TTL to 168 minutes.

The following stipulations apply to the *percentage_TTL* and *maximum_TTL* variables:

- ◆ The *percentage_TTL* and *maximum_TTL* variables are used only if a last-modified time is specified for the object.
- ◆ The *maximum_TTL* variable must be specified if the *percentage_TTL* is specified.
- ◆ When *percentage_TTL* is set to zero or is unspecified, NetCache uses the value specified for *absolute_TTL*.

How often objects are refreshed

About the IMS query interval

NetCache verifies objects for consistency based on the expiration time, or TTL, of the object, as described in “[How long objects remain in the cache](#)” on page 12. In this way, if the Web server has not included an explicit expiration time in the object itself, NetCache intelligently calculates how long the object should remain in the cache before sending an If-Modified-Since (IMS) query that verifies with the Web server that the object is still current.

Modifying the interval between IMS queries

You can define IMS rules for NetCache to increase or decrease the IMS query interval for a class of objects in order to improve response time for those objects, because objects are verified with their origin Web servers less frequently. However, some objects delivered to users might be stale. You can monitor the average response times on the HTTP - General page under the Data tab.

Regardless of an object’s lifetime setting or a NetCache TTL pattern applied to that object, NetCache verifies the consistency of an object whenever a defined IMS rule instructs NetCache to send an IMS query about that object. When the object is referenced by a user request, NetCache verifies the object if it has not been verified in the number of seconds specified in an IMS rule applied to that object.

The IMS rules table

The Header and Caching Behavior - IMS Rules tab page displays the IMS rules that determine the maximum time (in seconds) between object refreshes in this NetCache appliance.

No object can exceed this time between refreshes. Maximum age limits from the origin server that specify longer times are overridden by this setting.

This section lists the current IMS rules. IMS rules assign one or more forced refresh rates to objects based on time cached, source server, client, URL, or content type.

To view existing IMS rules in the Appliance Manager, select Setup tab > HTTP > Header and Caching Behavior. Click the IMS Rules tab to display the Current If-Modified-Since Rules table. This table lists the current If-Modified-Since rules.

How NetCache handles objects resulting from aborted requests

What is an aborted request?

An aborted request is a request made by a client and subsequently canceled by the client, for example, if the user clicks Stop or Reload in the Web browser.

How NetCache responds to a canceled request

By default, NetCache stops retrieving an object as soon as the client cancels the request. You might want this behavior if your users are served by a busy cache or if your users access the cache over a slow link, such as a SLIP or PPP link.

However, not completing the retrieval of an object can be undesirable. The reason is that when a client cancels a request for a large object, NetCache might already have retrieved the majority of the object. If NetCache stops retrieving the object when the client cancels the request, NetCache must retrieve the entire object later if another client requests it.

You can use the Complete Fetch option on the HTTP - Header and Caching Behavior page of the Appliance Manager to configure NetCache to continue the fetching and caching of an object even if the client cancels an object request before completion.

Determining whether to use the Complete Fetch option

If the Complete Fetch option is enabled, NetCache continues retrieving an object after the requesting client cancels the request. Impatient users can tie up resources if they repeatedly request and cancel objects that cannot be cached.

If your upstream bandwidth is expensive and you want to minimize its use, leave the Complete Fetch option disabled. If your bandwidth is not expensive, leave this option enabled so that NetCache can finish downloading the object for the next client that requests it.

Note

If NetCache is configured to operate in a dual mode as a Web accelerator and as a Web cache, leave the Complete Fetch option enabled to ensure that NetCache can finish retrieving the object.

Section C: General HTTP configuration options

About the General HTTP configuration options

To access the General HTTP configuration options in the Appliance Manager, select Setup tab > HTTP > General. You are not required to configure these options. The initial setup configuration and default settings are appropriate for most environments.

Enabling and disabling HTTP

You can cause NetCache to start or stop servicing HTTP traffic using the HTTP Enable check box on the HTTP - General page. This option is enabled by default.

There are certain conditions under which you might want to disable HTTP service. For example, you might not want to allow HTTP access to NetCache administrative functions, or you do not want the Web cache to service HTTP-encapsulated streaming requests.

Enabling authentication for HTTP requests

To cause NetCache to begin authenticating users for HTTP requests, select the Authenticate HTTP Requests option.

Make sure you have configured the authentication option you want to use before enabling this option. If you have not configured NetCache to use a specific type of authentication, users will not be able to use NetCache to service Web requests.

Select and configure the type of user authentication you want to use on the Authentication pages under the Setup tab in the Appliance Manager.

Note

If you want to require users to supply a user name and password when using Gopher, see the online Help associated with this protocol.

About the Client Connection Timeout option

You can cause the HTTP connection used for clients to drop after a specified time of inactivity. This option is enabled by default for 10 seconds and can be increased on the HTTP - General page.

About configuring the HTTP Proxy Ports option

The entries in the HTTP Proxy Ports option are the ports on which NetCache listens for client Web requests. However, you might need to configure this option differently, depending on which functions NetCache is performing on your network. These ports can be entered on the HTTP - General page.

Setting for Web cache: If you are configuring NetCache to run as a Web cache, you can leave the HTTP Proxy Ports option set to its default value of 3128. You can change the default value if you prefer a different port number to be used for client Web requests, or you can add more port settings to this option.

Setting for transparent proxying: If you are configuring NetCache to run as a transparent Web cache, add port 80 to the HTTP Proxy Ports option.

Additional options on the HTTP - General page

You can configure other options on the Appliance Manager HTTP - General page to specify how the NetCache appliance handles requests.

The options in the following table are available.

If you want to...	Use this option...
Configure the appliance to transparently redirect requests for URLs to specified local mirror sites	URLs to Rewrite
Configure the appliance to not send the client IP address when attempting to fetch content from a Web server	Client Privacy Enable
Suppress error messages that refer to the presence of a proxy-cache server to clients	Do Not Report Errors to Selected Clients
Suppress information about the presence of this appliance in the Via header for HTTP requests	Suppress Writing to Via Header Request
Suppress information about the presence of this appliance in the Via header for HTTP responses	Suppress Writing to Via Header Response
Configure the appliance to maintain a client connection beyond the servicing of a single request	Client-Side Persistence Enable

If you want to...	Use this option...
Configure the appliance to maintain a Web server connection beyond the servicing of a single request	Server-Side Persistence Enable

Section D: About FTP support

About this section

This section provides information about the support that NetCache provides for the File Transfer Protocol (FTP).

Topics in this section

This section includes the following topics:

- ◆ [“Overview of NetCache FTP support”](#) on page 22
- ◆ [“FTP object management”](#) on page 23
- ◆ [“Support for FTP connections”](#) on page 25

Overview of NetCache FTP support

What is FTP?

FTP is a method for moving files across a wide variety of computers. FTP uses TCP to move data. FTP enables a user to provide credentials to an FTP server and then list folders and transfer files in either direction. FTP also enables the client to initiate file transfer between the client and the FTP server.

FTP methods that NetCache supports

NetCache supports FTP as follows:

FTP requests over HTTP: FTP over HTTP supports Web browsers that request FTP URLs on the HTTP proxy port. Transparent FTP supports all FTP clients transparently, including Web browsers.

For FTP over HTTP, users must set up the FTP proxy in their Web browsers to point to the NetCache appliance or to an automatic proxy configuration file.

Transparent proxying of FTP requests: For transparent proxying, users do not have to configure their Web browsers to access the NetCache appliance. As a result, users are unaware of the existence of a NetCache appliance; that is, the appliance is *transparent*.

To set up transparent FTP, you must do the following:

- ◆ Configure the NetCache appliance for transparent FTP
- ◆ Configure an L4 or L7 switch or a WCCP router to redirect port 21 (FTP) requests to a NetCache appliance.

Be sure that you set up the redirection for FTP requests on port 21 and not on any other ports. All FTP commands are supported.

For information about setting up transparent FTP, see the transparent proxying chapter in the *Administration Guide*.

Nontransparent proxying of FTP requests : For nontransparent proxying, users must set up the FTP proxy in their Web browsers to use the FTP client. Nontransparent FTP provides proxy support for commonly used nontransparent/nonencapsulated clients, such as WS_FTP and CuteFTP, as well as native FTP clients for UNIX, Linux, and Microsoft Windows and DOS.

For information about setting up nontransparent FTP, see the nontransparent FTP topics in the Appliance Manager online Help, as well as the software documentation for the FTP client that you are using.

FTP object management

**FTP data that
NetCache caches**

FTP servers often serve large, cacheable documents. By caching these documents on the NetCache appliance and serving them from the cache, you can save a significant amount of bandwidth.

For transparent or nontransparent FTP: The following table shows the process that NetCache uses to ensure that the object is fresh.

Stage	Description	
1	NetCache checks the cache to determine if the object is in the cache.	
	If...	Then...
	The object is not in the cache.	NetCache fetches the requested object from the origin server and sends the object to the client. (END)
	The object is in the cache.	NetCache proceeds to Stage 2.
2	NetCache checks the last modified time on the object.	
	If...	Then...
	The last modified date of the object in the cache is the same as the last modified date of the object on the server.	NetCache serves the object in the cache to the requesting client. (END)
	The last modified date of the object in the cache is earlier than the last modified date on the object on the server.	NetCache fetches the requested object from the origin server and sends the object to the client. (END)
	No last modified date exists on the object.	NetCache proceeds to Stage 3.

Stage	Description	
3	NetCache checks the value of the Default Maximum Age option.	
	If...	Then...
	The object's age is less than the maximum age value.	NetCache serves the object from the cache to the requesting client. (END)
	The object's age is greater than the maximum age value.	NetCache fetches the requested object from the origin server and sends the object to the client. (END)

If the user reloads an object multiple times successively, NetCache fetches the object from the FTP server.

For FTP over HTTP: NetCache caches the objects fetched from the FTP server. NetCache uses the Default Maximum Age value to ensure the freshness of the FTP data in the cache.

Specifying data that is not to be cached

You can specify data that you do not want NetCache to cache, for example, an executable file (.exe). For information about configuring FTP caching, see the Appliance Manager online Help.

FTP Hierarchy

Hierarchies enable you to define what occurs when the appliance you are configuring cannot resolve a request; that is, what happens when a cache miss occurs.

With no hierarchy configured, the appliance forwards the request to the Internet and attempts to retrieve the requested object from the origin server and return it to the appliance. With a hierarchy configured, the appliance forwards the request to the next level up in the hierarchy, its parent cache. If the parent has it, it returns it to the appliance; if not, it forwards the request up to the next level. After reiterating this process through all configured levels, the request is finally forwarded to the Internet.

For information about configuring an FTP hierarchy, see the Appliance Manager online Help.

Support for FTP connections

About FTP connections

With FTP, the following two TCP connections are established between the client and the FTP server:

- ◆ A control connection over which FTP commands from the client and responses from the FTP server are transferred
For example, the client requests an object and the server acknowledges the request. Some FTP commands elicit multiple responses from the server.
- ◆ A data connection over which the server sends data to the client or the client sends data to the server
Files are transferred only over the data connection.

When a NetCache appliance is deployed, the NetCache appliance transfers commands on behalf of the client and receives the responses from the FTP server on behalf of the client.

Determining which device should establish the data connection

Before NetCache can retrieve a file from an FTP server (on behalf of the client), a data connection must be established with the FTP server. When you configure FTP support on NetCache, you specify how you want the data connection to be established.

Many administrators do not want the FTP server to establish the data connection to the NetCache appliance. The reason is that permitting a server outside their network to access devices on their network can pose a significant security risk. However, if server-side problems prevent the NetCache appliance from establishing a data connection to the FTP server, you need to consider allowing the FTP server to establish the connection to the appliance.

For example, if the FTP server responds positively to the NetCache request to establish the connection, but the FTP server does not accept connections to the IP address and port number that the NetCache appliance supplies, you might want to specify that the FTP server is to establish the connection. This situation might occur if an FTP server is behind a firewall that does not allow the connection requests to be passed through.

The optimal solution in this situation is for the administrator of the FTP server to reconfigure the system so that it allows the NetCache appliance to establish the connection to it. However, if you are unable to influence the FTP server's administrator to reconfigure the server, you can enable an option in the NetCache configuration so that NetCache can retrieve the desired object.

Note _____
If the firewall is set up to prevent FTP servers from establishing the connection to the NetCache appliance, specifying the option to allow the FTP server to establish the data connection prevents NetCache from proxying *any* FTP requests.

NetCache option for connection establishment

NetCache includes an option, FTP Connect by PORT, that enables you to specify whether NetCache is to establish data connections to FTP servers or whether NetCache will allow FTP servers to establish data connections to it. This option includes two suboptions, which are described in the following table.

Suboptions	Description
PASV, then Retry with PORT	This NetCache appliance will establish the connection to FTP servers. If the NetCache appliance is not successful, it sends a command to the FTP server to establish the connection to it. This is the default setting.
PORT Only	FTP servers will establish the connection to this NetCache appliance.

Location of the option: Select Setup tab > FTP > General in the Appliance Manager to access the FTP Connect by PORT option.

If the client uses the PORT command to establish a connection

The client can negotiate the data connection establishment with the NetCache appliance by using the FTP PORT and PASV commands. Even if the client establishes the connection to the NetCache appliance by using the PORT command, NetCache can still use the PASV command to establish the connection between the appliance and the FTP server. Your configuration of the FTP Connect by PORT option controls only how the connection will be established between the NetCache appliance and the FTP server.

Section E: About tunneling HTTP-based traffic

Overview of HTTP tunneling options

Several features are available to support tunneling of traffic directed to HTTP ports on the appliance. All options to support these features are on the HTTP - Tunneling page (Setup tab > HTTP > Tunneling).

Options to support CONNECT traffic

The following options are available for you to optimize handling of CONNECT requests:

- ◆ **Authenticate HTTP-Based Tunneling Requests option**
You can require user login and authentication for HTTP-based CONNECT requests to this appliance.
- ◆ **HTTP-Based Tunneling ACL option**
You can specify ACL rules specific to handling HTTP-based CONNECT requests to this appliance, for example, to allow server port 443 or 563 traffic.

Note

The Destination Ports option on the Network - General page can be used to deny access to nonstandard HTTP ports (for example, port 8080). However, Network Appliance recommends that you use the HTTP-Based Tunneling ACL option for that purpose. The reason is that using the Destination Ports option might be a more complex method of allowing traffic for nonstandard ports for protocols that do not pose the same security risks as HTTP.

Handling non-HTTP traffic on HTTP proxy ports

Some organizations use standard HTTP ports for non-HTTP traffic. Their L4 or L7 switches or WCCP routers are configured to redirect such traffic to the NetCache HTTP proxy port. Ordinarily, NetCache returns an error in such a case because it cannot recognize the traffic as HTTP traffic. However, you can use the Tunnel Non-HTTP Traffic option to instruct the NetCache appliance to accept non-HTTP traffic on its HTTP proxy ports and tunnel this traffic to the origin server.

This feature can be used only when HTTP transparent proxying is configured.

Tunneling traffic for specific server IP addresses

Some applications use traffic on port 80 that looks similar to HTTP but is not. NetCache ordinarily would return an error message in this circumstance. You can, however, configure NetCache so that NetCache tunnels all traffic destined to server IP addresses that you specify. NetCache then passes through the traffic to those servers through without examining it.

The options you can use to configure NetCache to tunnel this traffic are

- ◆ Enable Tunneling of Server IP Addresses
- ◆ Tunnel Server IP Addresses

This feature can be used only when HTTP transparent proxying is configured.

Section F: Web cache performance

About Web cache performance

The NetCache appliance is a high-performance system. However, if your environment places an extremely heavy load on an appliance, you can maximize system performance by doing the following:

- ◆ Change the NetCache configuration to improve hit rate.
- ◆ Disable some of the logging options.
- ◆ Add network interface cards.
- ◆ Modify the Cookie Cheat File Name Extensions list.
- ◆ Increase the object TTL.
- ◆ Examine the construction of and reduce the number of access control and caching rules.
- ◆ If your cache is busy or is running over a slow link, make sure the Complete Fetch option is enabled.
- ◆ Apply TCP Large Windows for long-haul connections.

Improving the hit rate of NetCache

The *hit rate* reflects the number of user requests that NetCache resolves by serving an object from the cache instead of fetching it from the origin server. The hit rate is a measure of the bandwidth savings that NetCache provides.

You can improve the NetCache hit rate using the following methods in the order of preference:

- ◆ Adding entries to the Cookie Cheat File Name Extensions list
- ◆ Increasing the Default Maximum TTL setting
- ◆ Adding disk space, if your cache has many large objects

Configuring for a high hit rate also improves overall system performance because NetCache is fetching fewer objects from the Internet, and is opening and closing fewer connections with origin servers.

The current status of HTTP traffic, hit-rate percentage, cached and uncached HTTP connections, and successful and failed HTTP connection attempts can be monitored on the HTTP - General page under the Data tab.

URL object hit rate versus URL byte hit rate: NetCache keeps statistics of both the percentage of objects and the percentage of bytes delivered from the cache. Because objects vary in size from very small to very large, these two hit-rate percentages typically differ.

If the hit rate seems low: You can view the object hit rate in the Current Status section of the HTTP - General page. To access this page, select Data tab > HTTP > General.

Alternatively, you can use a log-analysis utility to look for URLs that are referenced frequently but not cached (that is, they have a low hit-rate percentage). These objects, if cached, would enable you to increase the hit rate. After you identify some of these objects or types of objects, look at your NetCache configuration to determine why these objects are not cached.

Disabling the logging of FQDNs

The most significant performance improvement to be made with regard to logging is from the disabling of the logging of fully qualified domain names (FQDNs). Disabling this option does not affect which logs NetCache keeps; rather, it causes NetCache to log information by IP address instead of by domain name. DNS lookups are avoided.

Caution

If logging of FQDNs is disabled, access controls using domain names no longer work. If you decide to disable this log entry, check your access control entries and change any entries that use “client-domain” to use “client-ip” instead.

The logging of FQDNs can be disabled by using the Custom Log Format feature on the HTTP - Logging page under the Setup tab.

Adding one or more network interfaces

If the NetCache appliance has remaining CPU power and the current network interface is operating near its capacity, adding network interfaces increases performance.

You can see the system activity on a NetCache appliance by viewing the System Statistics - General page under the Data tab.

Modifying Cookie Cheat File Name Extensions entries

Web servers often use cookies to signify that a URL is user-specific with content that is inherently noncacheable. However, even when some of the content of a URL is user-specific, the URL might contain other objects that are static, such as

graphics files, ZIP files, and PDF files. NetCache enables you to define these objects as cacheable (even with a cookie), even when the surrounding content is user-specific and, therefore, noncacheable.

Placing only appropriate entries in the Cookie Cheat File Name Extensions option improves performance and increases hit rate because it enables NetCache to cache static objects that exist inside dynamic URLs. *This method is the one most likely to improve your hit rate.* If hit rate is important to you, include as many file types in this list as you can, keeping in mind that users might receive some objects that are no longer current.

You can add extensions to the Cookie Cheat File Name Extensions option on the HTTP - Header and Caching Behavior page under the Setup tab. Click the Help icon next to the Cookie Cheat File Name Extensions option to access help and the link to an extensive list of file name extensions. You can copy extensions from the Help file link, paste them into the Cookie Cheat File Name Extensions box, and edit them.

Increasing the object TTL

Increasing the TTL improves performance and increases hit rate because objects remain in the cache longer. However, objects delivered to users can be stale.

Sites interested in saving bandwidth can set object TTLs to several years; sites wanting total consistency can set object TTLs to weeks or hours. The more limited your wide-area bandwidth, the longer you should set your object TTLs.

You can establish or modify the TTL rules that NetCache uses for objects on the HTTP - Header and Caching Behavior page. In the TTL Patterns option, configure each object's TTL by entering TTL pattern rules.

Number of access control and caching rules

Keep only those rules required for your environment: The best general guideline to remember is that the more you configure NetCache to do, the slower the performance. Specifically, if NetCache is configured with many stoplists against which it must verify URL requests, you are trading some performance for access control.

Configure your access control options to ensure that NetCache performs only the checking that is required for your environment.

Avoid having long lists of regular expressions in your access control lists: Because regular expressions are the slowest rules to process, having long lists of regular expressions in your access control lists and URL control lists can reduce performance.

Use caution when setting rules: Variable expressions or rules apply a variety of restrictions. However, the type of rule and the number of rules that NetCache must check before servicing each request can affect performance. For example, rules containing the `server-ip` variable prompt a cached DNS lookup that might be slower than using the `server-name` variable to specify server names.

Options used to configure ACLs are located on the Access Control Lists page of the Appliance Manager. Rules for controlling access to protocols can also be applied from protocol-specific pages in the Appliance Manager, for example, the HTTP - General page.

For more information about ACLs, see the *Security Guide*.

Enabling the Complete Fetch option

With the use of the Complete Fetch option, you can specify whether NetCache should continue retrieving a requested object after the user cancels the request.

You might want to enable this option if your bandwidth is inexpensive and you want to maximize cache hits. By completing the retrieval of an object after the user aborts, the cache can serve that object to the next client that requests it without initiating a connection to the origin server.

Enabling the Complete Fetch option is also beneficial when your clients are served by a cache serving a heavy volume of requests, or if your users access the cache over a slow link (such as a SLIP or PPP link).

However, you should not enable the Complete Fetch option if your bandwidth is expensive. When the Complete Fetch option is disabled, you save bandwidth because NetCache stops retrieving objects as soon as the user aborts.

Not completing the retrieval of an object can be undesirable. When a client cancels a request for a large object, NetCache probably has retrieved the majority of the object. If NetCache stops retrieving the object when the client cancels the request, NetCache must retrieve the entire object again if another client requests it.

If your NetCache appliance is busy most of the time or is running over a slow link, make sure that the Complete Fetch option on the HTTP - Header and Caching Behavior page is enabled. The default setting is enabled.

Applying TCP large windows to long-haul networks

Enabling TCP large windows dramatically improves performance over high-bandwidth, long-haul connections by using the capacity of the pipe more effectively. Long-haul networks have a greater capacity than can be used

effectively with the standard TCP window size. Increasing the size of the TCP window enables NetCache appliances to communicate with hosts and subnets using the full capacity.

Using large windows on local area networks, however, has no benefit.

Networks that can benefit from enabling large windows include the following:

- ◆ NetCache appliances communicating with each other over a satellite link
- ◆ Clients connecting to a NetCache appliance using DSL but with a high-capacity hop between them and the appliance

You can use the TCP Large Windows Enable and the TCP Large Windows List options on the Network - General page to enable and specify the systems or subnets with which NetCache should communicate using large windows, and also to specify the window size that NetCache should use with each.

For details about the TCP Large Windows Enable and TCP Large Windows List options, see the routing chapter in the *Administration Guide*.

Section G: Monitoring a Web cache

About the Web access log

One of the available log files on the NetCache appliance is the Web access log. With the Web access log, the HTTP client transactions received by a Web cache, cache hits and misses, the types of objects, time required to handle objects, client IP addresses, bytes delivered, and other relative details to HTTP Web cache traffic can be recorded. For the field order and descriptions for the Web access log, see the chapter about log file formats in the *Guide to Messages and Log File Entries*.

NetCache log files and formats

When the NetCache appliance boots for the first time, all of the log files are enabled and set to a default format. However, you can modify the content and format of the logs.

About this chapter

This chapter provides planning information that will help you make the configuration decisions required to set up your appliance as a streaming media cache.

Before reading further

The information in this chapter assumes that you have read the streaming media information in the *Deployment Guide*. The *Deployment Guide* helps you understand

- ◆ What streaming media is
- ◆ The different ways NetCache can be configured for streaming media:
 - ❖ Streaming media cache
 - ❖ Streaming media acceleratorSee Chapter 3, “[Planning for Web or Streaming Acceleration](#),” on page 107 for information about operating NetCache as a streaming media accelerator.
- ◆ The ways you can deploy the appliance to meet your streaming media requirements

Sections in this chapter

This chapter contains the following sections:

- ◆ Section A, “[Planning for streaming media configuration](#),” on page 37
- ◆ Section B, “[Streaming media configuration requirements](#),” on page 44
- ◆ Section C, “[Customizing your streaming media configuration](#),” on page 61
- ◆ Section D, “[About streaming object management](#),” on page 80
- ◆ Section E, “[Troubleshooting the streaming media configuration](#),” on page 85
- ◆ Section F, “[Tuning a streaming media cache for performance](#),” on page 90
- ◆ Section G, “[Streaming media protocol details](#),” on page 95

Procedures and option descriptions

See the Appliance Manager online Help for configuration procedures and option descriptions related to streaming media.

Location of configuration options

To access the streaming media options in the Appliance Manager, select:

Setup tab > Streaming

Section A: Planning for streaming media configuration

About this section

This section provides information to help you plan to configure a NetCache appliance as a streaming media cache. This section also guides you through the steps that are required to plan the configuration of your NetCache appliance.

Topics in this section

This section provides information about the following topics:

- ◆ [“Planning for network deployment”](#) on page 38
- ◆ [“Determining the streaming media configuration requirements”](#) on page 39
- ◆ [“Additional features to consider implementing”](#) on page 40
- ◆ [“Applicability of other NetCache features to streaming”](#) on page 42

Accessing the Appliance Manager and the CLI

After you have planned how to configure your NetCache appliance as a streaming media cache, you can configure the appliance using the Appliance Manager (the NetCache browser-based administration interface) or the NetCache command line interface (CLI). For information about accessing the Appliance Manager, see the *NetCache Quick Setup* document. For information about the NetCache CLI, see the *Command Line Reference*.

Planning for network deployment

Decisions to be made

While you read the *Deployment Guide*, you made decisions about deploying your NetCache appliance as a streaming media cache on your network. Some of the deployment decisions you made require configuration using the Appliance Manager or the NetCache CLI.

The following table lists deployment decisions that require configuration, and it provides the location of information to help you plan how to configure the features that you want to implement.

If you want to...	See...
Deploy the appliance outside the firewall or with no firewall	◆ <i>Deployment Guide</i> ◆ Hierarchies chapter in the <i>Administration Guide</i>
Deploy the appliance inside a transparent or nontransparent firewall	◆ Hierarchies chapter in the <i>Administration Guide</i> ◆ “Basic firewall requirements for streaming media” on page 51
Deploy NetCache with one or more caches upstream	Hierarchies chapter in the <i>Administration Guide</i>
Use IP takeover pairs to enable one NetCache appliance to take over services for another if one fails	Takeover chapter in the <i>Administration Guide</i>
Deploy NetCache as a transparent cache with an L4 or L7 switch or WCCP 2.0-based router	Transparency chapter in the <i>Administration Guide</i>
Deploy NetCache as a nontransparent cache	◆ <i>Deployment Guide</i> ◆ “Implementing nontransparent redirection for MMS streaming” on page 65
Implement a load balancing strategy	◆ “Tuning a streaming media cache for performance” on page 90 ◆ Transparency chapter in the <i>Administration Guide</i>

Determining the streaming media configuration requirements

Decisions to be made

In this chapter, you will determine the following:

- ◆ Requirements for implementing transparent or nontransparent client access to NetCache
- ◆ Whether to use unicast or multicast transmission
- ◆ Port requirements for configuring NetCache, firewalls, and associated network devices for streaming media
- ◆ Implications of blocking streaming media ports on your firewall
- ◆ How to ensure that client logs are sent to streaming servers
- ◆ If you need to set up proxy authentication for streaming media
- ◆ Whether to enable HTTP service for streaming media
- ◆ How to
 - ❖ Limit the client request volume
 - ❖ Reduce redundant fetching
 - ❖ Implement nontransparent redirection for MMS
 - ❖ Prefill streaming media content
 - ❖ Use URLs to direct streaming media requests to the cache
 - ❖ Understand streaming media connection statistics
 - ❖ Manage streaming media objects

Information to gather for the configuration

Before continuing with NetCache configuration, ensure that you do the following:

- ◆ Obtain the Microsoft® Media Server (MMS), RealNetworks® RealSystem®, or Apple® QuickTime™ license codes required for streaming.
- ◆ Identify the port settings for the streaming media protocols you want to stream.
- ◆ If you are setting up nontransparent client access, record the IP address of the streaming media cache.
- ◆ Record the list of ports currently allowed on your firewall.
- ◆ Check the routers and switches in your network to ensure that they can handle the high-bandwidth demands of streaming media.

Additional features to consider implementing

Optional features available

NetCache includes a number of features that you can use to customize the NetCache appliance for your environment. Use the following table to help you determine which features to implement.

If you want to...	See...
◆ Customize the configuration of network interfaces ◆ Create virtual interfaces	Network interface chapter in the <i>Administration Guide</i>
Configure NetCache for host name resolution and for DNS caching	DNS chapter in the <i>Administration Guide</i>
Configure NetCache to use default and static routes for incoming and outgoing requests	Routing chapter in the <i>Administration Guide</i>
◆ Restrict administrative access for security purposes ◆ Require users to authenticate themselves before NetCache services requests ◆ Prevent unauthorized clients from accessing NetCache services ◆ Restrict the access of authorized clients to NetCache services	<i>Security Guide</i>
Restrict access to Web sites that contain content that is objectionable to your organization	Access control chapter in the <i>Security Guide</i>
Configure the SNMP agent on the NetCache appliance to use diagnostic and other management services	SNMP chapter in the <i>Administration Guide</i>
Select and modify log file formats	<i>Guide to Messages and Log File Entries</i>
◆ Enable or disable log files ◆ Select and modify log file formats	Log file chapter in the <i>Administration Guide</i>
◆ Customize error messages ◆ Hide error messages from users	Error handling chapter in the <i>Administration Guide</i>

If you want to...	See...
Control bandwidth use	The bandwidth allocation topic in the Appliance Manager online Help
Configure the ContentDirector software to distribute content from a central source to remote NetCache appliances	The ContentDirector agent topic in the Appliance Manager online Help

Applicability of other NetCache features to streaming

Settings for dedicated streaming media caches

If you do not want your streaming media cache to carry any traffic other than RTSP or MMS traffic, you should disable ports for other protocols.

Note

If you disable HTTP, the streaming media cache will not be able to respond to HTTP-encapsulated client requests for streaming media.

Web filtering settings might not always apply

Currently, the SmartFilter control list does not contain any restrictions for URLs beginning with the rtsp: or mms: prefix. This means that you cannot enable SmartFilter to specifically restrict access to these types of URLs. If clients enter an MMS or RTSP URL in their media players or click directly on a streaming media link, SmartFilter category restrictions will not apply.

However, if your clients' Web requests are directed to a Web cache, you can still block access to MMS or RTSP links by setting the SmartFilter categories on the Web cache to block the Web pages containing the restricted streaming media links.

In addition, you can restrict access to specific streaming media URLs by entering the appropriate rule in the Streaming ACL option. For example, the following access control rule denies access to the specified URL:

```
deny url rtsp://badsite.com/bad_content
```

Windows Media does not support the redirect access control rule

The redirect action rule does not apply to Windows® Media™ clients. Windows Media players do not respond to the 302 Moved Temporarily redirect message and do not reissue a connection request for the new location. Therefore, NetCache interprets the redirect action rule as “deny” for Windows Media.

The redirect rule works properly for QuickTime and RealNetworks clients; they are redirected to the URL specified in the redirect rule.

**User access and
cacheability rules
for HTTP-
encapsulated
streaming media**

When streaming media service is enabled, streaming requests using HTTP encapsulation are controlled by the same user access and caching behavior rules as all other streaming requests. The Streaming ACL option controls user access to streaming requests using HTTP encapsulation, and the options on the Splitting and Caching Behavior page (Setup tab > Streaming > Splitting and Caching Behavior) control how NetCache caches streams fetched for HTTP-encapsulated requests.

If Web service is enabled and streaming media service is disabled, HTTP-encapsulated streaming requests are controlled by the user access controls and cacheability options set for HTTP (Setup tab > HTTP).

ICAP not supported

The Internet Content Adaption Protocol (ICAP) is not supported for use with streaming media. Note, however, that the metafiles (.asx files, .ram files, and so on) that direct players to streaming content are HTTP objects and might pass through an ICAP server.

**Hierarchy
limitations**

Hierarchies are supported for streaming media caching. All hierarchy members handling MMS and RTSP must be identified as a parent. You cannot configure a streaming media cache as a neighbor. (A neighbor is a logical role that can be assigned to hierarchy members on a Web cache.)

See the hierarchy chapter in the *Administration Guide* for information about hierarchies.

Section B: Streaming media configuration requirements

About this section

This section describes the configuration requirements that you must meet to enable streaming media caching and splitting.

Topics in this section

This section provides information about the following topics:

- ◆ [“General streaming media requirements”](#) on page 45
- ◆ [“Configuration recommendations”](#) on page 47
- ◆ [“Basic firewall requirements for streaming media”](#) on page 51
- ◆ [“MMS streaming requirements”](#) on page 53
- ◆ [“RealSystem streaming requirements”](#) on page 54
- ◆ [“QuickTime streaming requirements”](#) on page 56
- ◆ [“Multicast requirements”](#) on page 58
- ◆ [“confill requirements”](#) on page 59

General streaming media requirements

License prerequisite

You must have a valid Windows Media™ (MMS), RealSystem, or QuickTime license code before setting up your NetCache appliance for streaming media.

You can obtain MMS, RealSystem, or QuickTime streaming media licenses from your Network Appliance™ sales representative or from the Installed Base Group (ibg@netapp.com).

Minimum switch firmware versions

For streaming media deployments, Network Appliance recommends that you use the following switch firmware versions:

- ◆ Foundry firmware version SLB7.1.02.bin or later for the Foundry ServerIron L4 switch. To obtain the latest Foundry firmware, contact Foundry technical support at support@foundrynet.com.
- ◆ Extreme firmware version 6.1.X for ExtremeWare. To obtain the latest Extreme firmware, contact Extreme technical support at support@extremenetworks.com.

Streaming media protocols supported by NetCache

NetCache supports streaming media for the following data types:

- ◆ Microsoft® Media Server (MMS) for Windows Media from Microsoft Corporation
- ◆ Real Time Streaming Protocol (RTSP) for RealSystem from RealNetworks®
- ◆ Real Time Streaming Protocol (RTSP) for QuickTime from Apple® Corporation for QuickTime Player and RealSystem

Each of the preceding media types requires a streaming license. See your Network Appliance sales representative for more information.

Note

If you do not purchase a streaming license, NetCache will not cache or split streaming traffic.

**Media player
versions supported
by NetCache**

NetCache supports the following media player versions:

- ◆ Microsoft Windows Media Player 6 and later
- ◆ RealNetworks RealPlayer® 6.1 (G2) and later
- ◆ Apple QuickTime Player 4 and later

Note

NetCache does not support the Application layer packet retransmission protocol introduced in QuickTime 5. Therefore, dropped packets are not retransmitted.

RealNetworks and QuickTime players are fully supported for caching and splitting. All other RTSP-based players are proxied by NetCache.

Configuration recommendations

Recommended strategy for initial configuration

Network Appliance recommends that you configure streaming media service as follows:

1. Configure associated network devices for the streaming media ports required for MMS or RTSP service
2. Bring up the NetCache appliance on the network
3. Configure streaming media on the appliance
4. Test and troubleshoot your streaming media configuration
5. Configure your streaming media object management settings

Use transparency to direct requests to NetCache

The recommended way to deploy NetCache with streaming media service is to use an L4 or L7 switch or a Cisco router running WCCP 2.0 and configure transparency on the streaming media cache and on the switch or router.

See the transparency chapter in the *Administration Guide* and the *Deployment Planning Guide* for more information about transparent proxying.

Note

If you plan to implement IP spoofing with transparency, plan your deployment carefully. IP spoofing deployments are complex, requiring a variety of planning and configuration tasks. Before you deploy IP spoofing, see the transparency chapter in the *Administration Guide*.

Enable HTTP service

Network Appliance recommends that you enable HTTP so that clients can fail over to HTTP transport for streaming media when other ports are blocked. If HTTP is disabled, your NetCache appliance will not listen for any Web traffic, including HTTP-encapsulated streaming media requests.

Recommended duplex settings

Network Appliance recommends that you configure all clients, caches, servers, and switches for full duplex to ensure proper data delivery. Streaming content across a network configured for half-duplex data delivery is not advisable.

Determining your duplex settings: To determine the duplex settings on the streaming media cache, enter the `ifconfig -a` command. The following example shows sample output of the `ifconfig -a` command:

```
e0: flags=848043<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 172.30.200.53 netmask 0xffffffff broadcast 172.30.200.255
ether 00:a0:98:00:4e:52 (auto-100tx-up)
```

In the preceding example, `auto-100tx-up` indicates that the streaming media cache's NIC is configured for auto-negotiated 100Base-T half duplex. If your switch's NIC setting shows “auto FD” (full duplex), you should change the streaming media cache settings to match the settings of your switch.

Changing the interface duplex settings: You can change the configuration of the streaming media cache's network interfaces using the Appliance Manager or the CLI. See the network interfaces chapter in the *Administration Guide* for interface configuration procedures.

Limit connections to reduce client request volume

If you are concerned that your client request volume might overload the NetCache appliance or the origin server, you can change the value specified in the MMS Maximum Client Connections option and the RealSystem Maximum Client Connection option. These options limit the number of MMS and RealSystem client connections that the NetCache appliance will service. The default setting for these options is 0 (zero), which means that NetCache will accept unlimited client connections.

Note

RealSystem and Windows Media multicast clients are included in maximum client connection totals.

Do not block streaming media ports on your firewall

Blocking UDP or TCP streaming ports can seriously degrade the quality of your streaming media. When streaming ports are blocked, media players (unless configured differently) attempt to contact the streaming server using less-efficient protocols.

Configure clients to use UDP

To achieve optimal playback, streaming media must be delivered on time and at a constant bit rate. Unlike HTTP clients and servers that wait long periods to deliver and receive data, streaming clients and servers begin thinning or

switching streams as soon as they encounter any type of network congestion. To speed data delivery and improve your users' playback experience, configure your streaming clients to use UDP for data transport.

Verify your UDP Port Range settings

In NetCache 5.1, the default port range for the UDP Port Range option was changed from 4000 to 65000 to 6970 to 65000. When you upgrade to NetCache 5.1 or later, the UDP port range is automatically changed to reflect the new defaults.

Note

Some servers deny access to clients that use low numbered UDP ports.

Ensure that clients are configured for Windows Media metafile rewriting

For Windows Media metafile rewriting to work, users must request streaming media content from within their browsers. The reason for this is that the Windows Media metafile rewriting process uses a nontransparent Web cache to intercept client HTTP requests for streaming media. Therefore, Windows Media metafile rewriting will not work if users access streaming media content directly from the media player (and not through a browser). The streaming request is never redirected to the streaming media cache because no initial HTTP request is sent to the Web cache.

See [“Implementing nontransparent redirection for MMS streaming”](#) on page 65 for more information.

Recommended methods of prefilling streaming content

You can use the following methods to prefill the cache with streaming media content:

- ◆ The `confill` command (derived from *content fill*)

You can use the `confill` command to download origin server files into the cache. The `confill` command downloads a streaming media file and caches it to disk so all clients can access it. You can use HTTP or FTP to download the source file. See [“Prefilling streaming media content”](#) on page 69 for more information about the `confill` command.

- ◆ ContentDirector

ContentDirector is a Network Appliance software application that enables you to upload files stored in a central location, such as a data center, to NetCache appliances located throughout the network, and then manage those files after they are distributed. See the ContentDirector documentation and

the ContentDirector Agent information in the online Help for information about using ContentDirector with NetCache.

Note

Noncacheable streaming content cannot be prefilled by either the `confill` command or ContentDirector.

**ACL rule creation
tips for RTSP
content**

Because RealPlayers append the port number to streaming URLs, your ACL rules will fail unless you include the port number in the URL string. For example, consider the following ACL rule:

```
deny url-prefix rtsp://foo.bar.org/
```

The RealPlayer will add a port number to any content at the preceding URL. For example, the URL `rtsp://foo.bar.org/baf.rm` will become `rtsp://foo.bar.org:554/baf.rm`.

Therefore, your ACL rule will not match and will fail.

Basic firewall requirements for streaming media

Streaming media configuration requirements for firewalls

The following table lists the basic tasks required for configuring transparent and nontransparent firewalls for streaming media.

Firewall type	Configuration requirements
Transparent firewall	Configure the firewall to allow the ports required for streaming media. No other specific Appliance Manager configuration is required.
Nontransparent firewall	Configure the firewall to allow the ports required for streaming media and include the firewall in the NetCache appliance hierarchy. To modify the request resolution hierarchy, configure the options on the Hierarchies pages in the Appliance Manager (Setup tab > Hierarchies). See the hierarchies chapter in the <i>Administration Guide</i> for more information.
Nontransparent firewall—IP multicast without tunneling	All requirements specified for nontransparent firewalls. In addition, verify that your network equipment supports multicast directly (without tunnels). For packet filtering purposes, the destination address range is 224.0.0.0 - 239.255.255.255.
Nontransparent firewall—IP multicast tunneling	All requirements specified for nontransparent firewalls. IP multicast tunneling is performed with IP-in-IP encapsulation. For packet filtering purposes, this is IP protocol number 4. To prevent a multicast tunnel from being used as a back door into or out of a network, verify that only multicast packets are allowed through the tunnel.

**Where to find
protocol-specific
firewall
requirements**

To learn about the ports that must be allowed for MMS, RealSystem, and QuickTime streaming, see the following sections:

- ◆ “[MMS streaming requirements](#)” on page 53
- ◆ “[RealSystem streaming requirements](#)” on page 54
- ◆ “[QuickTime streaming requirements](#)” on page 56

MMS streaming requirements

MMS port requirements for transparent redirection

To enable transparent proxying, you must configure the NetCache appliance, switches, or routers for MMS traffic. The recommended port numbers for transparent redirection of MMS traffic are as follows:

- ◆ TCP port 1755
- ◆ UDP port 1755
- ◆ TCP port 80 (for HTTP encapsulation or proxying)

Note—
The preceding port list includes all ports that must be redirected for multicasting.

Firewall requirements for MMS streaming

The following table describes the ports that should be opened to enable TCP and UDP MMS streaming traffic to pass through your firewall.

Note—
Although TCP port 80 is commonly used for HTTP streaming traffic, other ports, such as TCP 8080 and TCP 7070, can also be used.

Firewall location	Minimum suggested ports to open on firewall	Result if ports are not opened
Located between NetCache and the server	TCP 1755 TCP 80	If you do not allow TCP 1755, NetCache retrieves content using server-side HTTP encapsulation over port 80.
Located between NetCache and clients	TCP 1755 UDP 1755 TCP 80	If you do not allow TCP 1755, clients use TCP port 80 for client-side HTTP encapsulation.

See “[Streaming media protocol details](#)” on page 95 for descriptions of the protocols associated with the streaming media ports.

MMS multicasting support

Windows Media supports only scalable multicast streaming.

RealSystem streaming requirements

RealSystem port requirements for transparent redirection

- To enable transparent proxying, you must configure the NetCache appliance, switches, or routers for RealSystem (RTSP) traffic. The recommended port numbers for transparent redirection of RealSystem traffic are as follows:
- ◆ TCP port 554
 - ◆ UDP port 554
 - ◆ TCP port 80 (for HTTP encapsulation or proxying)
 - ◆ TCP port 8080 (for HTTP encapsulation or proxying)

Note_____

The preceding port list includes all ports that must be redirected for multicasting.

Firewall requirements for RealSystem streaming

- The following table describes the ports that should be opened to enable TCP and UDP RealSystem streaming traffic to pass through your firewall.
- Note**_____
- Although TCP port 80 is commonly used for HTTP streaming traffic, other ports, such as TCP 8080 and TCP 7070, can also be used.
-

Firewall location	Minimum suggested ports to open on firewall	Result if ports are not opened
Located between NetCache and the server	TCP 554 TCP 80 TCP 8080	If you do not allow TCP 554, NetCache uses TCP 80 for server-side HTTP encapsulation.
Located between NetCache and clients	TCP 554 UDP 554 TCP 80 TCP 8080	If you do not allow TCP 554, clients use TCP port 80 for client-side HTTP encapsulation.

See “[Streaming media protocol details](#)” on page 95 for descriptions of the protocols associated with the streaming media ports.

UDP restrictions for IP spoofing for RTSP

If you enable IP spoofing for RTSP, the RealPlayer® will not be able to use UDP and will either fail over to TCP (if the player has been configured to fail over) or fail. Therefore, if you must enable IP spoofing for RTSP, ensure that the RealPlayer failover option is enabled.

IP spoofing deployments are complex, requiring a variety of planning and configuration tasks. Before you deploy IP spoofing, see the transparency chapter in the *Administration Guide*.

RealSystem multicasting support

RealSystem supports back-channel and scalable multicast streaming; however, NetCache 5.2 and later supports only back-channel multicast streaming.

Be aware of RealSystem proxy splitting limitations

If NetCache is placed between a RealProxy™ and a RealSystem Server, NetCache will not split any live streams sent by the RealSystem Server. RealSystem Proxy, and OEM versions of RealSystem Proxy offered by RealNetworks' partners, use a special closed protocol to communicate with RealSystem Server. These connections use port 3030 for splitting connections and port 7802 for caching connections. Because NetCache does not listen on port 3030 or port 7802, these connections always bypass NetCache.

If RealProxy is unable to connect to the ports for these protocols, RealProxy will fail over to passthrough proxy mode for those streams accelerated by NetCache. In passthrough proxy mode, RealProxy does not offer any bandwidth savings. However, the upstream NetCache appliance will continue to perform splitting and caching.

To enable RealProxy to perform normally when accelerating a RealSystem Server (in nontransparent, reverse-proxy mode), list ports 3030 and 7802 in the Allowed Tunnel Destination Ports option (Setup tab > HTTP > Tunneling).

Note

Ports 3030 and 7802 are the default ports. You can configure RealSystem Server or RealProxy to use different port numbers for these connections.

If possible, configure the switch to redirect requests from the same client IP address to the same NetCache appliance.

QuickTime streaming requirements

QuickTime port requirements for transparent redirection

To enable transparent proxying, you must configure the NetCache appliance, switches, or routers for QuickTime (RTSP) traffic. The recommended port numbers for transparent redirection of QuickTime traffic are as follows:

- ◆ TCP port 554
- ◆ UDP port 2001
- ◆ TCP port 7070

Note

Redirecting port 7070 is recommended for streaming acceleration deployments only.

The preceding port list includes all ports that must be redirected for multicasting.

Open the UDP ports on the firewall

To stream QuickTime, you must open UDP ports on the firewall. These ports correspond to the port range listed in the UDP Port Range option in the Appliance Manager.

Firewall requirements for QuickTime streaming

The following table describes the ports that should be opened to enable TCP and UDP QuickTime streaming traffic to pass through your firewall.

Firewall location	Minimum suggested ports to open on firewall	Result if ports are not opened
Located between NetCache and the server	TCP 554 UDP <i>range</i> (destined for NetCache and in the port range specified in the UDP Port Range option) TCP 80 TCP 7070	If you do not allow TCP or UDP, NetCache uses TCP 80 for server-side HTTP encapsulation. NetCache caches and splits HTTP-encapsulated streaming content.

Firewall location	Minimum suggested ports to open on firewall	Result if ports are not opened
Located between NetCache and clients	TCP 554 UDP 2000 UDP 2001 TCP 80 TCP 7070	If you do not allow TCP 554, clients use TCP port 80 for client-side HTTP encapsulation.

See “[Streaming media protocol details](#)” on page 95 for descriptions of the protocols associated with the streaming media ports.

About TCP port 7070

To allow QuickTime HTTP encapsulation in a transparent proxying deployment, you must redirect TCP port 7070. However, TCP port 7070 is not included in the list of Well Known Ports (ports 0 to 1023) and might be used for other applications. Therefore, you should redirect TCP port 7070 only when you can ensure that the correct port traffic is being received, for example, in streaming acceleration or CDN deployments.

QuickTime multicasting support

Multicast streaming for QuickTime is not currently supported.

Multicast requirements

NetCache multicast streaming features

You can configure NetCache to convert unicast and multicast requests and streams using the following methods:

- ◆ **Multicast to unicast**—A content provider broadcasts in multicast, which NetCache converts to unicast and delivers to clients.
- ◆ **Unicast to multicast**—NetCache receives a unicast stream, which NetCache converts to multicast and broadcasts to clients in a multicast-enabled network.
- ◆ **Multicast to multicast**—NetCache receives a live stream from a server-defined multicast channel and sends the stream to clients through another NetCache-defined multicast channel.

Note

Currently, the multicast-to-multicast method applies only to RealPlayer connections.

IP address requirements in a multicast streaming environment

The range of multicast output IP addresses designated on the NetCache appliance must not be the same as any other NetCache appliances or users. Your organization might already have a range of IP addresses reserved for multicast streaming. If not, NetCache provides default addresses.

Port requirements in a multicast streaming environment

The range of multicast output ports designated on the NetCache appliance must not be the same as any other NetCache appliances or users. Your organization might already have a range of ports reserved for multicast streaming. If not, NetCache provides default port numbers.

confill requirements

About the confill command

You can use the `confill` command to download origin server files into the cache. The `confill` command downloads a streaming media file and caches it to disk so all clients can access it. You can use HTTP or FTP to download the source file. The following sections describe the use of the `confill` command. For a syntax description of the `confill` command, see “[Prefilling streaming media content](#)” on page 69.

Streaming media types supported by confill

You can use the `confill` command to pull the following types of streaming media content into the cache:

- ◆ Windows Media
- ◆ RealSystem
- ◆ QuickTime

QuickTime confill requirements

Consider the following when using the `confill` command to pull QuickTime files into the cache:

- ◆ You can prefill only QuickTime audio and video tracks.
- ◆ You cannot use the `confill` command to prefill QuickTime files that contain data in external files. All meta data and payload data must be present in the file that is to be pulled.
- ◆ For optimal performance, hint QuickTime streams using a “optimize for server” option. In the QuickTime Player, the option is called Optimize Hints for Server. In typical applications, this type of hinting improves `confill` performance.

confill requirements for acceleration deployments

If you are using the `confill` method to prefill content, you must use the URL of the origin server as the target URL of the `confill` command. Otherwise, your clients cannot access the prefilled content. The reason for this is that content must be cached under the URL of the origin server so that cache misses can be resolved from the origin server.

For example, the following acceleration rule correctly directs the streaming accelerator to search for content that includes the URL of the origin server:

```
accelerator-ip 1755 * origin-server-ip 1755 *
```

Given the preceding rule, if you prefill (cache) content using the IP address of the streaming accelerator, clients will not be able to access content. For example, consider content that is prefilled as follows:

```
rtsp://accelerator-ip/foo.rm
```

Because the acceleration mapping rules direct the appliance to look for content cached under the IP address of the origin server, your prefilled content will never be found.

Section C: Customizing your streaming media configuration

About this section This section describes additional streaming-related features that you can use to meet your deployment requirements.

Topics in this section This section provides information about the following topics:

- ◆ “[Reducing redundant fetching](#)” on page 62
- ◆ “[Implementing nontransparent redirection for MMS streaming](#)” on page 65
- ◆ “[Using URLs to direct requests to NetCache](#)” on page 67
- ◆ “[Prefilling streaming media content](#)” on page 69
- ◆ “[Ensuring that client logs are sent to servers](#)” on page 71
- ◆ “[Authenticating streaming client requests](#)” on page 74
- ◆ “[Planning for client request volume](#)” on page 75
- ◆ “[Understanding streaming media connection statistics](#)” on page 76

Reducing redundant fetching

Streaming content can be redundantly fetched

Because clients can access live or on-demand streaming content using disparate URL forms, NetCache might fetch the identical streaming content multiple times. For example, consider the following URLs:

`rtsp://mediaserver.lab.netapp.com/movie.rm`

`rtsp://mediaserver.lab/movie.rm`

`rtsp://mediaserver/movie.rm`

`rtsp://xxx.xxx.xxx.xxx/movie.rm`

Typically, requests for each of these URLs will result in four separate requests for streaming content. These redundant fetches have the following consequences:

- ◆ Additional upstream bandwidth is consumed to fetch streams that could be served by splitting a live stream or serving an on-demand stream from the cache.
- ◆ Each redundant fetch for on-demand streaming content is cached independently. For example, if the content had been previously cached when `rtsp://mediaserver.lab/movie.rm` was requested, a subsequent request for the same content from `rtsp://mediaserver/movie.rm` will result in a cache miss and a new connection to the origin server.

URL normalization reduces redundant fetching

You can use the streaming URL normalization options to eliminate the redundant fetches caused by URL differences. When you enable these options, NetCache resolves streaming URLs to find the authoritative domain name. NetCache will then split or cache the content using the Fully Qualified Domain Name (FQDN). Resolving URLs to the FQDN increases the cache hit rate because disparate URLs can be resolved to the same cached content.

When to use the streaming URL normalization options

Generally, the streaming URL normalization options are most useful in intranet deployments where many URLs might not contain an FQDN.

About the Streaming URL Normalization options

The following streaming URL normalization options are available:

- ◆ **Ignore URL Query String**
When the Ignore URL Query String option is enabled, NetCache ignores the segment of the URL string that appears after the “?” query for the purposes of deciding whether the stream should be split or served from the cache.
The original URL (with the query string) is sent to the origin server for logging and validation purposes.
- ◆ **Resolve IP Addresses to FQDN**
When the Resolve IP Addresses to FQDN option is enabled, NetCache performs a reverse DNS lookup on the IP address and caches the content using the FQDN. Resolving the IP address to the FQDN eliminates the possibility of caching duplicate content when a host has multiple IP aliases.
- ◆ **Resolve Host Names to FQDN**
When the Resolve host names to FQDN option is enabled, NetCache performs a DNS lookup on all streaming requests that contain nonqualified host names and resolves them to the FQDN.

Note

MMS URLs are always normalized to lowercase and have their trailing slashes (up to the first “?”) removed.

Before enabling the Ignore URL Query String option

If the query string includes information used to identify content and you enable the Ignore URL Query String option, NetCache will ignore the query string and retrieve the identical stream for every user accessing a given URL. Therefore, ensure that your streaming content is not distinguished by information included in the query string.

DNS considerations for streaming URL normalization deployments

DNS lookups for URL normalization can have the following consequences:

- ◆ **DNS lookup latency**
Because of the DNS lookup involved in finding the FQDN, the Streaming URL normalization options impose additional latency at connection time. However, the additional latency is normally offset by the improved hit rate caused by URL normalization and the performance improvements created by DNS caching.

◆ Media player timeouts

If any of the following conditions exists, the media player will be forced to sit idle and might eventually time out while NetCache waits for a DNS response:

- ❖ No DNS server exists.
- ❖ The DNS server fails.
- ❖ The DNS query fails.

Implementing nontransparent redirection for MMS streaming

About Windows Media metafile rewriting

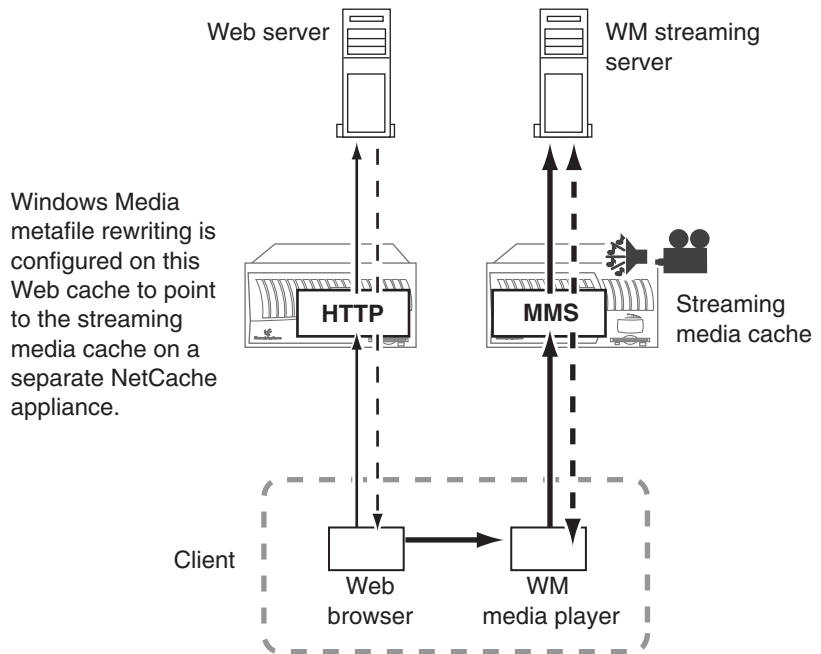
If you are not deploying transparent proxying for MMS, and your clients do not have Windows Media players that support proxy settings, you can direct requests to the streaming media cache by setting up Windows Media metafile rewriting.

Windows Media metafile rewriting provides a nontransparent way for you to direct MMS and other Windows Media type requests to the streaming media cache. Windows Media metafile rewriting works only with Windows Media streams from Windows Media servers.

To implement Windows Media metafile rewriting, your clients' Web requests must first be directed to a NetCache appliance configured as a Web cache. You configure the Web cache to add the IP address of the streaming media cache to the Windows Media pointer returned in a metafile by the Web server. All Windows Media requests are then directed to the streaming media cache instead of the streaming server.

Windows Media metafile rewriting example

The following example illustrates Windows Media metafile rewriting. In this example, one NetCache appliance has been configured as a Web cache and another NetCache appliance has been configured as a streaming media cache.



Note that client requests for streaming media originate from within the Web browser and are redirected to the NetCache appliance configured as a Web cache.

The request flow shown in the preceding example can be explained as follows:

1. A client initiates a Web request for streaming media.
2. The Web cache forwards these streaming requests to the Web server.
3. The Web server returns a streaming URL to the Web cache.
4. The Web cache inserts the IP address of the streaming media cache into the beginning of the streaming URL and returns it to the client.
5. Because of the metafile rewriting, the request for streaming media is redirected to the streaming media cache.
6. If the content is not cached, the streaming media cache fetches the content from the origin server and delivers it to the client.

Using URLs to direct requests to NetCache

About client access with URL modification

You can direct MMS or RTSP client requests to NetCache by providing clients with streaming URLs that contain the IP address (or host name) of the streaming media cache and the IP address of the streaming server. For example:

```
mms://cache/server/file  
rtsp://cache/server/file
```

How URL modification might be used

URL modification is especially useful in the following situations:

- ◆ If you want to dynamically generate URLs to direct clients to the closest or least loaded streaming media cache and streaming server
- ◆ When you want to keep clients from directly accessing the streaming server

Example of URL modification to direct clients to specific caches

A content distributor that owns both the streaming media cache and streaming media server could create a CGI script to dynamically generate URLs that direct clients to the streaming media cache and streaming server that have the lightest load. For example:

```
rtsp://least-loaded-cache/least-loaded-server/file.rm
```

The preceding URL directs the client to the streaming media cache with the lightest load. The streaming media cache then requests the content from the streaming server identified as having the lightest load.

Example of URL modification to isolate streaming servers from clients

Content distributors that do not want to provide clients with direct access to their internal streaming servers can provide URLs of the following form on their external Web site:

```
rtsp://external-cache/internal-server/file.rm
```

The preceding URL directs clients to the external streaming media cache. After receiving the client request, the streaming media cache then accesses the internal streaming server. To prohibit clients from accessing any other servers on the internal network, the content distributor could then construct ACLs that limit client access to only specified servers.

Description of URL modification syntax

Consider the following syntax example:

```
rtsp://cache/server/file
```

In the preceding example, *cache* is the IP address or host name of the streaming media cache, *server* is the IP address or host name of the streaming server, and *file* is the name of the streaming media content.

When the streaming media cache receives a client request for a URL of this type, it strips off the *cache* portion (IP address or host name of the streaming media cache) of the URL to find the IP address of the streaming server. The streaming media cache then requests content from the streaming server.

URL modification for HTTP progressive download

For URL modification and HTTP progressive download of streaming media to work properly, you must configure additional options. For more information, see “Setting up URL modification for HTTP progressive download” in the streaming media section of the Contents tab in the online Help.

Prefilling streaming media content

Prefilling described You can download streaming content into the cache so that clients can access streaming media content from within their own network (called *prefilling*). Prefilling the cache ensures that each cache contains the same streaming media content. An additional benefit is that streams are delivered quickly and with high quality because they are stored closer to clients.

Use the confill command to prefill Use the CLI `confill` command to prefill the cache. The `confill` command downloads an origin server file and caches it to disk so all clients can access it. You can use HTTP or FTP to download the source file. If you use HTTP, the origin server must be an HTTP server and not a streaming server using HTTP to stream the file.

Content fill example To pull content into the cache, you must enter the `confill` command and the source URL and target URL. For example:

```
confill http://foo.com/video.rm rtsp://foo2.com/video.rm
```

In the preceding example, the source file located at `http://foo.com/video.rm` is cached to disk with the URL `rtsp://foo2.com/video.rm`.

FTP example: The following example shows content fill using FTP:

```
confill ftp://username:password@foo.com//video.rm  
rtsp://foo2.com/video.rm
```

Syntax of the confill command The command syntax for the `confill` command is as follows:

```
confill [-l {<lock_age> | inf}] [-m {<max_age | inf}] <source URL>  
<target URL>
```

The following text describes the elements in the preceding syntax example:

-l <lock_age> : Specifies the period, in seconds, that the inserted streaming media file will be stored in the cache as a nonejectable object.

-l inf: Specifies infinite lock age. The inserted streaming media file will remain in the cache until manually ejected by the `eject` command. See “[Ejecting locked objects from the cache](#)” on page 70 for more information.

-m <max_age>: Specifies that the cache will never verify with the origin server that the inserted cached content is current (although it will contact the origin server to resolve misses).

-m inf : Specifies that the cache will never verify with the origin server that the inserted cached content is current (though it will contact the origin to resolve misses).

<source_URL>: Required element that specifies the source URL for the inserted streaming media content. This variable indicates the server that has the streaming media content to be inserted into the cache.

<origin_server_URL>: Required element that specifies the target URL. The target URL is the URL from which clients will request content from. Clients that request streaming content from this URL are served the inserted streaming media content.

Ejecting locked objects from the cache

The `eject` command can be used to eject any content from the cache, including content locked by the `confill` command. Ejected content cannot be used to satisfy client requests.

The syntax of the `eject` command is as follows:

```
eject <target_url>
```

For example:

```
eject rtsp://foo2.com/video.rm
```

The preceding command ejects the content associated with the following streaming media URL:

```
rtsp://foo2.com/video.rm
```

Ensuring that client logs are sent to servers

**Content providers
can still receive
client logging
information**

Many streaming media content providers require that clients send logging information back to the streaming server so that information can be collected for billing, advertisement revenue, performance analysis, and other purposes.

If you want to ensure that client logging information is forwarded to the streaming server, you must configure your NetCache appliance to do so by setting the log forwarding options on the MMS and RTSP pages in the Appliance Manager (Setup tab > Streaming).

Note

Even if you disable the MMS or RTSP log forwarding options, NetCache sends standard logging information to the streaming server.

**Client log
forwarding**

When client requests are routed through more than one streaming media cache, you must enable the client log forwarding options on each streaming media cache in the same way. If you enable client log forwarding on some caches but not others, client logging information will not be delivered to the streaming server. By default, client log forwarding options are turned on.

For example, assume that you have two streaming media caches between the client and the streaming server—one at a POP (point of presence) and another at a data center. Assume further that all POP traffic is routed through the data center. If the streaming media cache at the POP has log forwarding enabled but the streaming media cache at the data center does not, client logs will be forwarded from the POP to the data center, but no further. You must enable log forwarding on the streaming media cache at the data center to ensure that NetCache can forward client logs to the streaming server.

**Information
contained in client
logging information**

The following table shows the information contained in the logs that clients send to streaming servers. Even if a log does not record a client IP address, you can still determine the popularity of streams because all logs contain one entry line per client.

Logging information	RealMedia	QuickTime	Windows Media
URL	Yes	Yes	Yes
Client IP address	No	No	Yes
Play time	Yes	Yes	Yes
Connection time	Live media: Yes On-demand media: No	Yes	Yes
One entry line per client	Yes	No	No
Viewing quality statistics	Yes	No	Yes

Real-time client monitoring

You can configure the NetCache Real-Time Monitoring Protocol (RTMON) to monitor client streaming media activity and have NetCache send real-time statistics to application servers (for example, to billing servers). You can, for example, use the statistics to determine user demographics and popular URL access in order to determine load balancing. With NetCache deployed near clients, you can acquire information that other proxies block, such as original client IP addresses.

To use RTMON, an application server developer must program the interface on the server to the NetCache API for RTMON. Additionally, the appliance must be configured for RTMON. You can configure RTMON on the appliance on the Real-Time Monitoring- RTMON page in the Appliance Manager (Setup tab > Real-Time Monitoring > RTMON).

See *Client Monitoring and Control for Streaming* and the Application Manager online Help for details about real-time monitoring with RTMON.

Limiting log entry disk space requirements

Use the Maximum Buffer Size Per Server IP option (Setup tab > On Demand Statistics) to limit the amount of client event data that Netcache writes to disk. Setting a maximum event buffer size is useful if you think that your logging servers will not be able to keep up with the event data sent by clients. If you do not specify a buffer size and your logging servers cannot handle the data or are unavailable, NetCache will write all client event data to disk as long as disk space is available.

When you specify a maximum buffer size, NetCache writes client event data to disk until the buffer size is reached. See the Maximum Buffer Size Per Server IP option in the online Help for more information.

The default setting is 0 KB (unlimited event buffering).

Authenticating streaming client requests

Proxy authentication for streaming clients

You can force proxy authentication for streaming clients by enabling the Authenticate (MMS and RTSP) Requests option on the Streaming - General page in the Appliance Manager (Setup tab > Streaming > General). The Streaming (RTSP and MMS) Authentication option performs proxy authentication using the authentication settings entered in the Authentication pages of the Appliance Manager (Setup tab > Authentication).

The following authentication methods are supported:

- ◆ NetCache appliance user database
- ◆ LDAP
- ◆ RADIUS
- ◆ NTLM

End-to-end authentication

If a streaming server requires client authentication, NetCache establishes a connection to the streaming server and forwards the authentication information provided by the client. NetCache waits until the streaming server verifies the authentication request before delivering the stream to the client.

NetCache forwards the authentication information for *every* client—even if a client's request can be resolved by splitting an existing stream already authenticated for another client or by serving a stream from the cache. In these cases, NetCache opens a connection to the streaming server and forwards the client authentication credentials.

If NetCache subsequently delivers a split stream to the authenticated client, NetCache might close the connection to the streaming server after the client has been authenticated. On-demand connections are not terminated.

Planning for client request volume

Limiting client connections to NetCache

If you are concerned that your client request volume might overload the NetCache appliance or the origin server, you can change the value specified in the MMS Maximum Client Connections option, RealSystem Maximum Client Connections option, and QuickTime Maximum Client Connections option. These options limit the number of MMS, RealSystem, and QuickTime client connections that the NetCache appliance will service. The default setting for these options is 0 (zero), which means that NetCache will accept unlimited client connections.

Note

RealSystem and Windows Media multicast clients are included in maximum client connection totals.

Understanding streaming media connection statistics

Why streaming media statistics are more complex

Streaming media connection statistics can be more difficult to decipher than HTTP statistics. Unlike HTTP connections, streaming media connections often require more than one connection to the streaming server. For the most part, these multiple connections consist of data, control, and feedback connections. However, other connections can be initiated for a variety of reasons during streaming. This section describes the types of connections you might see and explains when these connections are initiated during streaming.

Where to find streaming media statistics

You can get streaming media statistics by accessing the following areas of the Appliance Manager.

Logging information: The following pages enable you to view streaming logging information:

- ◆ Log Files - Streaming Access
- ◆ Log Files - Streaming Details

To access these pages in the Appliance Manager, select Data tab > Log Files.

Statistical information: The following pages enable you to view streaming-related statistics:

- ◆ Streaming - General Statistics page
- ◆ Streaming - Detailed Statistics page
- ◆ Streaming - Server Statistics page
- ◆ Streaming - Hits and Misses page
- ◆ Streaming - Live Volume page
- ◆ Streaming - On-Demand Volume page
- ◆ Streaming - Proxy Volume page

To access these pages in the Appliance Manager, select Data tab > Streaming.

When UDP transport is used for client connections

NetCache supports UDP transmission of streaming data from the streaming media cache to the client. Many streaming media administrators prefer to use UDP transport because it is more efficient. Whether UDP is used, however, depends on whether the client media player is configured to use UDP and whether the firewall is configured to allow UDP ports.

Why you might see TCP connections to servers

Clients can request TCP, UDP, or HTTP data transport from NetCache. Whenever possible, NetCache uses TCP to connect to streaming servers. Using TCP to connect to streaming servers guarantees that packet loss will not interfere with the quality of streaming content.

How client connections are determined

Media players attempt to connect to streaming servers using the following transports and ports, in order of precedence.

Media player	Ports requested
Windows Media	UDP 1755 TCP 1755 HTTP 80
RealNetworks	UDP 554 TCP 554 HTTP 8080 HTTP 80
QuickTime	TCP 554 and UDP 2001 HTTP 7070 HTTP 80

Reasons that NetCache opens new server connections

- Even when NetCache serves content by splitting a stream or serving it from the cache, NetCache might open new connections to the streaming server to
- ◆ Verify the freshness of the stream
When a user requests a stream that is in the cache but has expired, that is, it has exceeded its Time-To-Live (TTL), NetCache verifies the freshness of the stream with the originating streaming server. See [“How NetCache handles streaming objects”](#) on page 83 for more information.
 - ◆ Send client logging information back to the streaming server

When NetCache is configured to send client logging information to the streaming server, NetCache opens a connection to the streaming server to provide information to the streaming server, for example, how many clients viewed the content, how many bytes were sent to the client on behalf of the streaming server, and the amount of the time that the client played the content.

- ◆ Establish a control connection to honor licensing requirements
NetCache must initiate a connection to the streaming server if such a connection is required to honor a vendor's licensing requirements. These connections are persistent throughout client playtime.
- ◆ Facilitate end-to-end authentication
If the streaming server requires end-user authentication, NetCache opens a connection to the streaming server to provide client authentication credentials.

These server connections are very low bandwidth and are usually initiated only during connection initiation and termination.

Connection termination under high memory load

When the appliance is under high memory load, it terminates the following types of connections to conserve memory:

- ◆ Connections that are fetching data from an origin server (to conserve memory)
- ◆ Live splitting connections

The appliance tries to avoid terminating connections that are serving data directly from the cache and connections that are just being proxied (these connections use less memory).

If you continually see the following entry in the Streaming details log, you might need to upgrade to an appliance with more capacity:

```
SERVER_DATA_PRIVATE_LOW_MEMORY
```

Multicast server connection termination

In scalable multicast streaming, NetCache cannot rely on client close-connection requests to terminate a unicast connection to the origin server. Instead, NetCache monitors traffic on each multicast channel. If, after a fixed amount of time, NetCache does not detect any Internet Group Management Protocol (IGMP) traffic for a particular multicast channel, NetCache tears down its connection to the origin server and terminates the current streaming session.

NetCache behavior for client logging connections

How NetCache forwards client logging information depends on whether the stream has been proxied, cached, or split.

Log forwarding for proxied streams: When NetCache proxies streams, it simply forwards any logging information sent by the client.

Log forwarding for streams served from the cache: When NetCache streams data from the cache, NetCache opens a control connection to the streaming server and forwards any logging information sent by the client.

Log forwarding for split streams: When NetCache delivers a split stream to a client, NetCache forwards the client logging information over the shared origin server connection. The association between individual client and server connections is not maintained.

Note

Some logging information might be blocked by going through the appliance, for example, the client IP address. To support client IP logging and access control at the server, you must implement IP spoofing. For more information, see the transparency information in the *Administration Guide*.

Dynamic stream switching can cause new connections

Because of the highly variable nature of available network bandwidth, the media player might need to increase or decrease the bit rate of the stream to achieve optimum playback. Because streams are differentiated by bit rate, NetCache must provide the client with a new stream every time the client requests a different bit rate. This stream switching can occur several times as the user plays streaming content.

Each time the media player requests a new bit rate during playback, NetCache delivers the new stream using one of the following methods:

- ◆ Splitting an existing stream if one is available (live content)
- ◆ Delivering a cached stream if one is available (on-demand content)
- ◆ Fetching a new stream from the streaming server if NetCache cannot satisfy the request by splitting a stream or delivering a cached stream

NetCache can dynamically switch streams and not affect client playback.

Section D: About streaming object management

About this section

This section defines cacheable and noncacheable streaming objects, describes how NetCache handles streaming objects, and explains the different methods you can use to manage streaming objects.

All users should read this section.

Topics in this section

This section provides information about the following topics:

- ◆ [“Cacheable streaming media content”](#) on page 81
- ◆ [“How NetCache handles streaming objects”](#) on page 83

Cacheable streaming media content

NetCache is preconfigured to cache content

After you install a Windows Media, RealSystem, or QuickTime streaming media license and enable streaming media, the NetCache appliance is ready to cache on-demand streams using the default caching behavior settings. You can modify the way streams are cached using the options on the Splitting and Caching Behavior page in the Appliance Manager (Setup tab > Streaming > Splitting and Caching Behavior).

Cacheable streaming content

Generally, NetCache caches RealNetworks, QuickTime, and Microsoft on-demand streaming media content. Live streams are split. See the *Deployment Guide* for more information.

Noncacheable streaming media content

The following streaming media content is not cacheable:

- ◆ All RealNetworks non-RealMedia content:
 - ❖ SMIL (Synchronized Multimedia Integration Language)
 - ❖ RealText®
 - ❖ RealPix™
 - ❖ RealFlash
- ◆ RealMedia content with over 15 encodings in a particular track
- ◆ RealSystem Server authenticated streams
- ◆ RealNetworks streams that do not use unicast RDT (Real Data Transport) as the data transport
- ◆ Windows Media content accessed in fast-forward or rewind mode
- ◆ QuickTime content with unknown RTP timing parameters

Cacheable RealMedia MIME types

NetCache will split or cache the following RealMedia MIME types:

- ◆ audio/x-pn-realaudio
- ◆ video/x-pn-realvideo
- ◆ application/x-pn-realevent

The application/x-pn-imagemap and image_map/x-pn-realvideo MIME types are not supported. These MIME types are proxied by NetCache.

**Prefilling
restrictions**

Noncacheable streaming content cannot be prefilled by either the `confill` command or ContentDirector.

**Other content types
that NetCache will
not cache or split**

NetCache will not split or cache any of the following content:

- ◆ Managed content whose minimum age has not yet elapsed
- ◆ Content types for which the NetCache appliance has no license (for example, if only QuickTime is licensed, you cannot split or cache RealNetworks media).

How NetCache handles streaming objects

NetCache performs segmented caching of streams

NetCache caches streams by dividing them into fixed time segments. Each stream is cached as a series of objects, each containing n seconds of data. When a user plays a stream, NetCache caches each segment of streaming media as soon as it has played for enough time to be segmented.

If a client watches five minutes of a stream, NetCache caches as many objects as it can segment in that five minutes. If another client subsequently requests the same content, NetCache serves the cached content and prefetches portions of content that were not cached when the client is within n seconds of viewing it.

How streaming content is stored on disk

Because NetCache stores streams as a collection of discrete objects, entire streams do not have to be ejected from the cache when disk space is required to cache a small Web object. For example, a small portion of a large stream can be ejected to make room for new HTTP or streaming objects.

About streaming Time-To-Live

Streaming protocols do not have a formalized notion of Time-To-Live (TTL).

Standard RTSP streams: The default TTL for RTSP streams is equal to the length of the stream. For example, a two-hour film has a default TTL of two hours.

RealNetworks streams: To comply with RealNetworks licensing, the NetCache appliance revalidates RealNetworks streaming content each time the content is requested.

MMS streams: The default TTL for MMS streams is found by comparing the length of the stream to 20 percent of the last modified time. The lesser of these two values becomes the default TTL. For example, if the stream length is two hours and the stream was last modified one day earlier, the TTL for the stream will be 2 hours (because 20% of 24 hours = 4.8 hours).

Note

If the time specified in the Force Refresh Rate option (Setup tab > Streaming > Splitting and Caching Behavior) has expired, NetCache revalidates the stream, regardless of the stream's TTL. See the online Help for more information.

Modifying the default TTL

You can modify the TTL behavior for both RTSP and MMS by locking the content using the `confill` command or by modifying the value specified in the Force Refresh Rate option (Setup tab > Streaming > Caching and Splitting Behavior).

For more information about the `confill` command, see “[Prefilling streaming media content](#)” on page 69.

What happens when a stream expires

When a stream expires, NetCache does not immediately invalidate the object in the cache. Instead, all versions of all streams remain in the cache until NetCache needs to reclaim disk space. If a client requests a stream of one format, such as RTSP, and the same stream is cached in another format, such as MMS, the MMS stream is not ejected.

When an expired stream is referenced by a user request, NetCache sends a query to the originating streaming server to determine whether the stream in the cache is the current version, that is, whether the stream has been modified. If the stream on the streaming server has not been modified, NetCache serves the cached stream to the user and resets the stream’s cache timestamp. If the stream on the streaming server has been modified, NetCache caches the new version and delivers it to the client.

Section E: Troubleshooting the streaming media configuration

About this section This section describes common problems you might have while deploying your streaming media cache and provides troubleshooting recommendations.

If you are experiencing poor streaming performance The following table lists the actions you can take to improve your client’s playback experience.

Corrective action	Where to find more information
Purchase a streaming license that allows more bandwidth	“If you suspect that throughput is a problem” on page 86
Check your auto-negotiation and duplex settings	◆ “If clients are buffering, switching to lower bit rates, or failing” on page 86 ◆ “If you cannot determine the cause of network congestion” on page 86
Revert your network to a simple configuration	“If you cannot determine the cause of network congestion” on page 86
Check your hardware	◆ “If reverting your network configuration did not improve streaming” on page 86 ◆ “If you suspect your switch is the problem” on page 87 ◆ “If you suspect your switch is the problem” on page 87
Configure all network devices (including firewalls) for UDP traffic	“If you are experiencing excessive packet loss” on page 87
Check RealPlayer settings	◆ “If RTSP clients are failing” on page 87 ◆ “If clients are failing in an IP spoofing deployment” on page 88

Corrective action	Where to find more information
Verify that no other clients are monopolizing UDP	“If clients are buffering” on page 88
Check the appliance configuration and media properties	“If NetCache is not caching or splitting content” on page 88

If you suspect that throughput is a problem

Verify that you have a streaming license installed that allows the expected performance. Three levels of MMS and RealSystem licensing are available—Standard, Pro, and Ultra. These licenses are throughput-based and are available for the C2100 only.

If clients are buffering, switching to lower bit rates, or failing

If you configure the streaming media cache’s NIC for auto-negotiation, ensure that any connecting devices are also configured for auto-negotiation. If some devices are not configured for auto-negotiation or do not auto-negotiate properly, your duplex settings will be inconsistent and network conditions will deteriorate. See [“If clients are failing in an IP spoofing deployment”](#) on page 88 for duplex configuration details.

If you cannot determine the cause of network congestion

If you are experiencing packet loss and client failures and cannot determine the source of the problem, revert the network to a basic configuration. Configure clients, servers, streaming media caches, and switches to 100Base-T full duplex and test client playback. This simple configuration should enable you to incrementally increase the complexity of your network configuration so that you can determine the problem.

If reverting your network configuration did not improve streaming

If you have reverted your network to a clean configuration and still cannot determine the problem, the problem might lie with faulty hardware. Check your cables and connections and replace any hardware that you suspect might be faulty.

If you are experiencing excessive packet loss

Network configurations that use the NetCache appliance's Gigabit Ethernet NIC to deliver streaming data to 100Base-T clients can be problematic. If you have a switch with a Gigabit Ethernet interface connected to the NetCache appliance and a 100Base-T interface connected to clients, the switch will drop packets if the buffer cannot manage the traffic. This appears to be a limitation of current switch technology.

If you suspect your switch is the problem

To determine if a switch is the cause of your network difficulties, collect the port statistics to determine if the switch is dropping packets. If the number of packets entering the switch is greater than the number of packets leaving the switch, the switch is dropping packets. To fix the problem, test a different type of switch on your network.

If you are experiencing difficulties with streaming QuickTime traffic

If you want to stream Apple QuickTime content using the UDP transport protocol and your firewall blocks UDP traffic, you must configure the firewall to allow UDP traffic to the NetCache appliance for the port range specified in the UDP Port Range option.

The UDP Port Range option is located on the Streaming - RTSP page in the Appliance Manager (Setup tab > Streaming > RTSP).

If you do not allow UDP traffic, your clients will attempt to use HTTP. To allow HTTP-encapsulated QuickTime streaming, you must allow TCP port 80 and TCP port 7070.

If RTSP clients are failing

PNA (Progressive Networks Audio) is a RealNetworks proprietary protocol used for backwards compatibility with RealSystem 3.0 through 5.0. The NetCache streaming media caching and splitting features do not support the use of PNA. Therefore, ensure that the PNA proxy setting in the RealPlayer is not configured to point to the streaming media cache.

Note

NetCache proxies all HTTP-encapsulated PNA requests.

If clients are failing in an IP spoofing deployment

If you enable IP spoofing for RTSP, RealPlayers will not be able to use UDP and will either fail over to TCP (if the player has been configured to fail over) or fail. Therefore, if you must enable IP spoofing for RTSP, ensure that the RealPlayer failover option is enabled.

If clients are buffering

When testing your streaming media configuration, ensure that no other clients are using UDP to receive data. If other clients on the network are using UDP, they will monopolize the available bandwidth in congested networks.

If clients are failing to receive HTTP encapsulated streams

Using HTTP encapsulation for streaming media creates multiple GET and POST connections between the client and the NetCache appliance. If the deployment includes a switch for load balancing, there is no guarantee that all requests from the same player will be redirected to the same NetCache appliance. This means that some of the requests the player makes while playing a stream could be sent to the wrong NetCache appliance. The client, expecting a request, drops the connection, and the occurrence is not logged.

If NetCache is not caching or splitting content

If the NetCache appliance is not caching or splitting content, verify that

- ◆ Your Streaming Cacheability Substring option (Setup tab > Streaming > Splitting and Caching Behavior) values are not prohibiting caching.
- ◆ Your hierarchy rules are not preventing splitting or caching.
- ◆ The RTSP media does not contain the following headers:
 - ❖ Pragma: no-cache
 - ❖ Cache-Control: private
 - ❖ Cache-Control: no-cache
- ◆ The media has an X-MEIPort header.
- ◆ The RealNetworks streams do not require RealServer authentication. RealSystem Server authenticated streams cannot be cached. For more information, see [“Noncacheable streaming media content”](#) on page 81.

If you see abnormal connection termination

When the appliance is under extreme memory load, it will terminate connections that are fetching data from origin servers and live splitting connections to conserve memory.

Note

The appliance will try to avoid terminating connections that are serving data directly from the cache and connections that are just being proxied (these connections use less memory).

The following message in the Streaming details log indicates that connections are being dropped because of memory pressure:

```
SERVER_DATA_PRIVATE_LOW_MEMORY
```

To rectify the problem, you might need to upgrade to an appliance with more capacity.

Section F: Tuning a streaming media cache for performance

About tuning and load balancing

This section contains information regarding tuning by load balancing when you are operating your NetCache appliance as a streaming media cache with a WCCP 2.0 router or an L4 switch.

It also provides instructions on gathering streaming media data and interpreting the statistics to gauge the performance of a streaming media cache.

Importance of load balancing data over network interfaces

You cannot achieve optimal performance unless you first ensure that your network deployment properly load balances connections across the network interfaces of your NetCache appliance.

Streaming media proxies can deliver streaming data to clients very quickly over multiple network interfaces. However, if a proxy is forced to deliver streaming data only over a single interface, the proxy is likely to become limited in its network throughput by the bandwidth of the single network interface.

To ensure proper load balancing, do both of the following:

- ◆ To load balance client connections, deploy the streaming media proxy with a router running WCCP 2.0 and configure the WCCP NetCache IP Addresses for Load Balancing option on the Transparency - WCCP Global Settings page under the Setup tab of the Appliance Manager.
- ◆ To load balance server connections, configure the Load Balancing Interfaces option on the Network - General Settings page under the Setup tab of the Appliance Manager.

Note

Network Appliance recommends that you configure both the WCCP NetCache IP Addresses for Load Balancing option and the Load Balancing Interfaces option.

Load balancing with WCCP 2.0: When you deploy your NetCache appliance with a router running WCCP, you can enable NetCache to deliver data over multiple network interfaces by setting the WCCP Load Balancing option on the Transparency Settings page; you list the NetCache network interfaces over which you want NetCache to deliver data to clients. If you do not list any network interfaces, NetCache delivers data to clients over a single interface, which NetCache chooses.

Use the following guidelines to determine which, if any, network interfaces to list in the WCCP Load Balancing option:

- ◆ List only IP addresses for network interfaces through which this NetCache appliance communicates with clients. Do not, for example, list a network interface through which the NetCache appliance communicates with another NetCache appliance.
- ◆ When using server-side load balancing and WCCP load balancing, make sure that the WCCP load balancing list is consistent with the server-side load balancing list if the NetCache appliance uses the WCCP router as a default gateway to servers. If the lists are not consistent, requests initiated by the appliance using an IP address that is not in the WCCP load balancing list might be returned to the appliance in a routing loop.
- ◆ Typically, NetCache can deliver HTTP and NNTP data over a single network interface without adversely affecting performance.

Load balancing using the Load Balancing Interfaces option:

By listing the IP addresses of *all* the network interfaces of your NetCache appliance in the Load Balancing Interfaces option on the Network - General Settings page, you enable NetCache to send out requests with different source IP addresses. This configuration ensures that Web and streaming server requests and responses are distributed over multiple network interfaces.

Note

Configuring the Load Balancing Interfaces option is beneficial if you expect the NetCache appliance to handle a large number of on-demand streaming requests.

Load distribution using a Foundry ServerIron L4 switch: If you are using a Foundry ServerIron L4 switch, you must configure the switch to ensure an even distribution of client connections across the cache interfaces by using round-robin or hashing on the client (source) IP address.

Check your firewall configuration

Ensuring that the NetCache appliance interacts correctly with the firewall is of paramount importance in establishing and maintaining streaming media quality. You can avoid many streaming media problems by ensuring that you have properly set up access to the firewall and have opened the required ports for streaming media.

The following list describes the effectiveness of each transport:

- ◆ UDP
UDP is the recommended transport for steaming media, and streaming media quality is highest when UDP is allowed. However, all UDP streaming

connections require additional TCP control connections. Therefore, to implement UDP streaming, you must also open the streaming TCP ports on the firewall.

- ◆ TCP
The quality of your streaming media degrades when TCP is used for data transport. However, you do not have to open UDP ports to enable TCP streaming. (QuickTime streaming requires UDP ports.)
- ◆ HTTP encapsulation
Streaming media quality is poorest when HTTP encapsulation is used for data transport. NetCache supports both server-side HTTP encapsulation and client-side HTTP encapsulation.
- ◆ HTTP proxied
If you do not have a streaming license or streaming is disabled, NetCache will proxy streaming traffic over HTTP (TCP port 80). When NetCache proxies streaming requests over HTTP, NetCache does not split or cache the traffic. From a bandwidth perspective, HTTP proxying is the least desirable method of transporting streaming media.

Note_____

See the *Command Line Reference* for category descriptions, syntax, and examples of CLI commands.

Using streaming media statistics to gauge performance

The Streaming - General Statistics page and the Streaming - Detailed Statistics page under the Data tab in the Appliance Manager provide traffic statistics about streaming real-time information, and streaming media connections and performance. You can customize the statistics reported in the Statistics tab page by selecting or clearing fields in the Options tab.

The following table shows some of the performance problems indicated by the statistics on the Streaming Media Instrumentation page.

If this is the case...	Then...
The General Statistics page shows a large number of delayed connections.	The NetCache appliance is heavily loaded and you should take immediate action to decrease its load.

If this is the case...	Then...
The Bytes Queued statistics show a large number of queued bytes.	A network problem exists. Check the client statistics to determine whether this problem is client-related. If the problem is not client-related, use <code>sysstat</code> to check if CPU capacity is at 100 percent or if network interfaces are at maximum throughput. Verify that data is properly load-balanced across all network interfaces.
The Packets Delayed statistics show that clients are dropping bytes.	The network or client is experiencing problems. Check the Client Bytes Received statistics and Client Bytes Sent statistics to see if a particular client, or group of clients, is dropping bytes.
The client information statistics indicate that more than 5 percent of clients are dropping packets.	A network or client problem exists. Check the Streaming access log to see if a particular client, or group of clients, is affected. To see streaming statistics, use the <code>sysstat -S</code> command.

CLI commands you can use to gather streaming data

The following CLI commands can be used to gather data about the performance of the streaming media cache:

- ◆ `sysstat`
Provides statistics indicating the current CPU usage, and the amounts of network and disk I/O.
- ◆ `sysstat -p`
Provides statistics about packets.
- ◆ `sysstat -S`
Provides streaming statistics indicating the current system load.
- ◆ `show status.streaming.*`

Provides statistics for streaming connections. These statistics are identical to those provided in the Appliance Manager.

- ◆ `netstat -an`

Displays the current network conditions.

- ◆ `netstat -p tcp`

Displays the current TCP statistics. The cache must be rebooted to clear TCP statistics.

- ◆ `ifstat -a`

Provides statistics about the interfaces enabled on the NetCache appliance.

- ◆ `smstats`

Provides statistics for active streaming connections.

Section G: Streaming media protocol details

About this section

Understanding the purpose and characteristics of streaming media connections is crucial to properly configuring your network for streaming media traffic. If you do not configure firewalls and other associated devices for the streaming media ports, your clients might not be able to receive content. This section provides detailed background information

Clarification of terms

This section uses the terms *upstream* and *downstream*. *Upstream* traffic flows from clients to NetCache and from NetCache to streaming servers. *Downstream* traffic flows from streaming servers to NetCache and from NetCache to clients.

Topics in this section

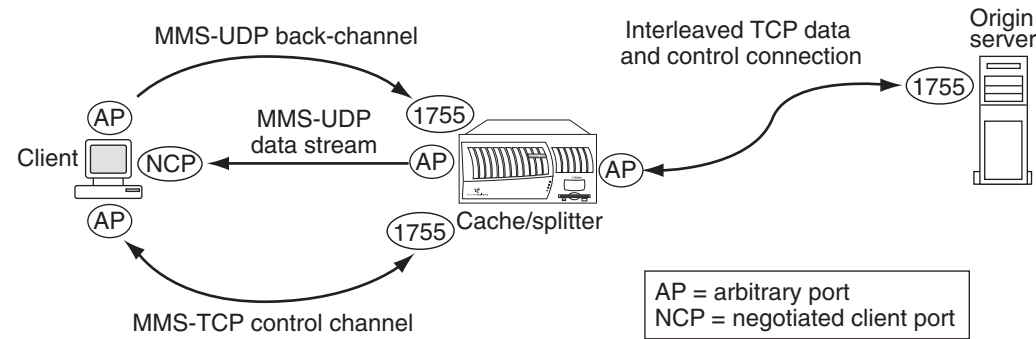
This section provides information about the following topics:

- ◆ [“MMS streaming media protocol details”](#) on page 96
- ◆ [“RealSystem streaming media protocol details”](#) on page 98
- ◆ [“QuickTime streaming media protocol details”](#) on page 100
- ◆ [“Understanding HTTP for streaming”](#) on page 102
- ◆ [“NetCache behavior for MMS multicast streaming”](#) on page 104

MMS streaming media protocol details

MMS streaming example: UDP to clients

The following example shows the typical client and server connections required for MMS streaming. Note that in this example, NetCache receives content from the streaming server over TCP and delivers the content to clients over UDP.



If a firewall did not allow TCP port 1755, the NetCache appliance would use HTTP encapsulation over TCP port 80 to send data to servers or clients. This scenario is not illustrated.

Understanding MMS connection requirements

The following table describes the types of protocols NetCache must service to deliver MMS content to clients.

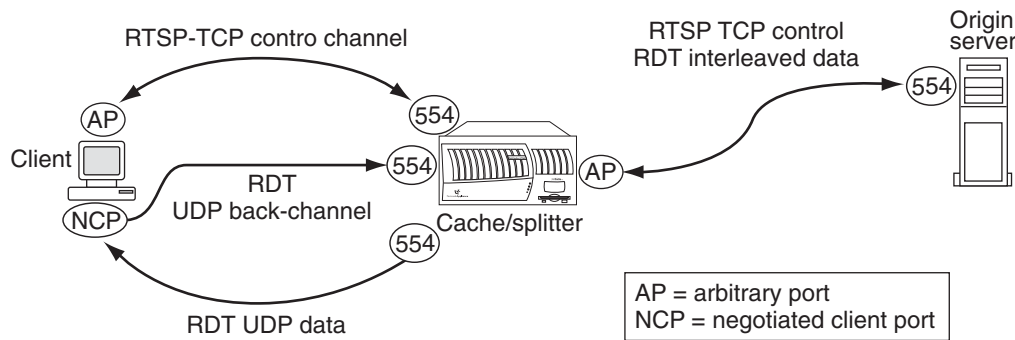
Use the proceeding table to configure the NetCache appliance and associated network devices, such as a switch, router, or firewall, that must pass traffic to and from the NetCache appliance. To ease configuration, the Appliance Manager provides default settings that correspond to the port information provided in the following table.

Protocol	Transport	Explanation	Upstream traffic	Downstream traffic
MMS	TCP	Microsoft Media Streaming. Used for data and control in Windows Media streams only.	Received on NetCache port 1755 Sent to port 1755	Received on arbitrary NetCache port Sent from NetCache port 1755 to arbitrary client port
MMS	UDP	Microsoft Media Streaming. Used for data and retransmit requests in Windows Media streams only.	Received on NetCache port 1755 Sent from arbitrary client port	N/A Sent to arbitrarily advertised UDP ports
HTTP	TCP	HTTP encapsulation is used from NetCache to clients and NetCache to servers when the firewall blocks TCP or UDP.	Received on NetCache port 80 Sent to origin server port 80	Received from origin server port 80 or 3128 Sent from NetCache port 80 or 3128

RealSystem streaming media protocol details

RealSystem streaming example: UDP to clients

The following example shows the typical client and server connections required for RealSystem streaming. Note that in this example, NetCache receives content from the streaming server over TCP and delivers the content to clients over UDP.



If a firewall did not allow one or more of the required ports, the NetCache appliance would use HTTP encapsulation over TCP port 80 to send data to servers or clients. This scenario is not illustrated.

Understanding RealSystem connection requirements

The following table describes the types of protocols NetCache must service to deliver RealSystem content to clients.

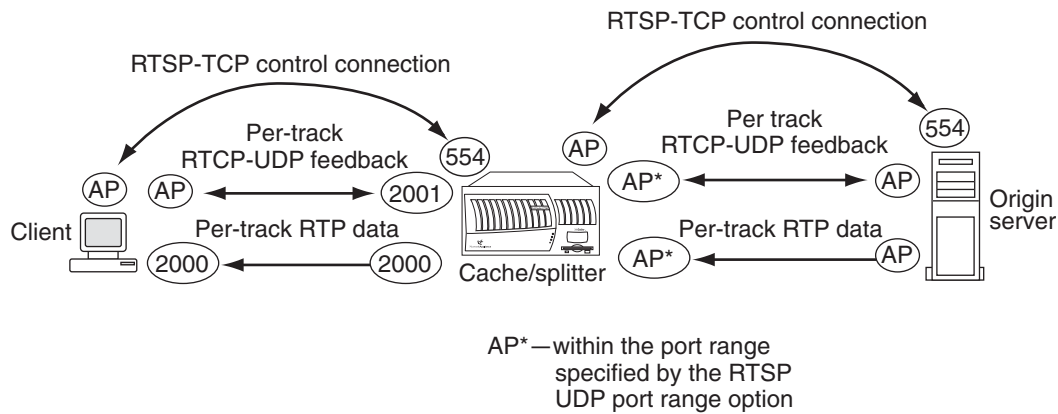
Use the proceeding table to configure the NetCache appliance and associated network devices, such as a switch, router, or firewall, that must pass traffic to and from the NetCache appliance. To ease configuration, the Appliance Manager provides default settings that correspond to the port information provided in the following table.

Protocol	Transport	Explanation	Upstream traffic	Downstream traffic
RTSP	TCP	Real Time Streaming Protocol. Used to coordinate the delivery of streaming media.	Received on NetCache port 554 Sent to port 554	Received on arbitrary NetCache port Sent from port 554 to arbitrary client ports
RDT or PN-TNG	UDP	Real Data Transport. RealNetworks proprietary protocol used for monitoring and feedback.	Received on NetCache port 554 Sent to arbitrarily advertised UDP ports	Received on NetCache port range 6970 to 65000 Sent from NetCache port 554 to arbitrarily advertised UDP ports
RTP	UDP	Real-Time Transport Protocol. Used for data transport for all proxied RealNetworks streams.	N/A	Received on NetCache port range 6970 to 65000 Sent from NetCache port 2000 to arbitrarily advertised UDP ports
HTTP	TCP	HTTP encapsulation is used from NetCache to clients and NetCache to servers when the firewall blocks TCP or UDP.	Received on NetCache port 80 Sent to origin server port 80 Can also be sent from or received on port 8080	Received from origin server port 80 or 3128 Sent from NetCache port 80 or 3128

QuickTime streaming media protocol details

QuickTime streaming example

The following example shows the typical client and server connections required for QuickTime streaming.



If a firewall did not allow one or more of the required ports, the NetCache appliance would use HTTP encapsulation over TCP port 80 to send data to servers or clients. This scenario is not illustrated.

Understanding QuickTime connection requirements

The following table describes the types of protocols NetCache must service to deliver streaming media content to clients.

Use the proceeding table to configure the NetCache appliance and associated network devices, such as a switch, router, or firewall, that must pass traffic to and from the NetCache appliance. To ease configuration, the Appliance Manager provides default settings that correspond to the port information provided in the following table.

Protocol	Transport	Explanation	Upstream traffic	Downstream traffic
RTSP	TCP	Real Time Streaming Protocol. Used to coordinate the delivery of streaming media.	Received on NetCache port 554 Sent to port 554	Received on arbitrary NetCache port Sent from port 554 to arbitrary client ports
RTCP	UDP	Real-Time Control Protocol. Provides quality of service information for QuickTime streams.	Received on NetCache port 2001 Sent to arbitrarily advertised UDP ports	Received on NetCache port range 6970 to 65000 Sent from NetCache port 2001 to arbitrarily advertised UDP ports
RTP	UDP	Real-Time Transport Protocol. Used for data transport for all QuickTime streams.	N/A	Received on NetCache port range 6970 to 65000 Sent from NetCache port 2000 to arbitrarily advertised UDP ports
HTTP	TCP	HTTP encapsulation is used from NetCache to clients and NetCache to servers when the firewall blocks TCP or UDP.	Received on NetCache port 80 Sent to origin server port 80 Can also be sent over port 7070	Received from origin server port 80 or 3128 Sent from NetCache port 80 or 3128

Understanding HTTP for streaming

Why you should enable HTTP

Although HTTP is not normally associated with streaming media, HTTP can be used as a streaming transport when other ports are blocked. If a client cannot connect to the streaming server using UDP or TCP, the client will try to connect using HTTP.

NetCache handles HTTP streaming traffic in the following ways:

- ◆ Client-side HTTP encapsulation
- ◆ Server-side HTTP encapsulation
- ◆ HTTP proxying

When NetCache uses client-side HTTP encapsulation

HTTP encapsulation is the process by which a streaming media protocol is tunneled over HTTP requests.

When a client attempts to contact the streaming server using HTTP, NetCache intercepts the request and attempts to fetch the content from the streaming server using TCP or HTTP (if the content is not cached). If NetCache can retrieve the content using TCP, NetCache then encapsulates the streaming data into HTTP for delivery to the client.

NetCache performs client-side HTTP encapsulation for Windows Media, RealNetworks, and QuickTime clients.

When NetCache uses server-side HTTP encapsulation

If NetCache is behind a firewall that blocks TCP or UDP traffic, or if the streaming server is only listening on an HTTP port, NetCache retrieves streaming content using HTTP encapsulation. When NetCache uses server-side HTTP encapsulation to retrieve streaming content, NetCache caches or splits the content and attempts to send it to clients using UDP, TCP, or HTTP (in that order).

NetCache performs server-side HTTP encapsulation for Windows Media, RealNetworks, and QuickTime clients.

HTTP proxying

If you have not installed a streaming license or if streaming is disabled, NetCache will proxy streaming traffic over HTTP. NetCache does not cache or split HTTP-proxyed streaming traffic.

Clients using HTTP encapsulation are controlled by the same user access and caching behavior rules as all other streaming requests. See “[User access and cacheability rules for HTTP-encapsulated streaming media](#)” on page 43 for more information.

NetCache behavior for MMS multicast streaming

About MMS multicast streaming

In a unicast environment, when a client makes a request (clicks an HTTP link) for an MMS stream that is not cached, NetCache connects to the origin server through the default MMS port. In a multicast environment, a live stream request does not result in a connection to the origin server. Depending on the configuration, NetCache responds as described in the following paragraphs.

Multicast in, unicast out: NetCache intercepts the multicast request at the HTTP proxy and performs an ASX rewrite to insert a special unicast request. NetCache then redirects the request to the designated streaming cache (if it is different) through the default MMS port. NetCache fetches the .nsc file, decodes it, and begins to receive multicast data from the media server.

For example, the original ASX request (initiated through a browser) is as follows:

```
http://web-server/foo.asx
```

The original ASX response is as follows:

```
<asx>
  <ref href = "http://web-server/foo.nsc">
</asx>
```

After the ASX rewrite, the MMS request is as follows:

```
<asx>
  <ref href = "mms://cache-ip/?http://web-server/foo.nsc">
  <ref href = "http://web-server/foo.nsc">
</asx>
```

Unicast in, multicast out: NetCache intercepts the multicast ASX response that redirects the subsequent .nsc request to a streaming server. NetCache assigns a new multicast channel (IP address and port) to this multicast request and modifies the .nsc response to reflect the new IP address and port number. NetCache then connects to the media server using the unicast URL embedded in the .nsc file and broadcasts the multicast stream.

For example, the original ASX request (initiated through a browser) is as follows:

```
http://web-server/foo.asx
```

The original ASX response is as follows:

```
<asx>
  <ref href = "http://web-server/foo.nsc">
</asx>
```

After the ASX rewrite, the response is as follows:

```
<asx>
  <ref href = "http://cache-ip/web-server/foo.nsc?">
  <ref href = "http://web-server/foo.nsc">
</asx>
```

Multicast Output Only: When this option is enabled, all streaming clients are forced to access live streaming content using the multicast protocol. You should enable this option if you want to ensure that only multicast delivery is used for live content streamed over your network.

Note

The Multicast Output Only option limits only client-side traffic to multicast transport. The server-side transport can be unicast or multicast.

About this chapter

This chapter describes configuring the appliance as a *Web accelerator* or a *streaming accelerator*. It contains a planning section, configuration overview, and configuration examples.

For an overview of the appliance deployed as an accelerator, see the *Deployment Guide*.

Before reading further

This chapter assumes that you have read the information about Web acceleration in the *Deployment Guide* and have made decisions about how you want to deploy your appliance as a Web accelerator on your network.

Sections in this chapter

This chapter includes the following sections:

- ◆ Section A, “[Planning for accelerator configuration](#),” on page 108
- ◆ Section B, “[Overview of accelerator configuration](#),” on page 113
- ◆ Section C, “[Accelerator configuration scenarios](#),” on page 127

Procedures and option descriptions

See the Appliance Manager online Help for configuration procedures and option descriptions related to Web and streaming acceleration.

Location of configuration options

To access the Web acceleration options in the Appliance Manager, select:

Setup tab > HTTP > Web Server Acceleration

Section A: Planning for accelerator configuration

About this section

This section describes the planning tasks that you should perform before configuring your appliance as an accelerator.

Topics in this section

This section provides information about the following topics:

- ◆ [“Gathering preconfiguration information”](#) on page 109
- ◆ [“Planning for uninterrupted service”](#) on page 112

Gathering preconfiguration information

You must plan the configuration

Accelerator configuration will require less time if you make some decisions and gather some information before you start configuring the appliance. To plan for acceleration configuration, you must do the following:

- ◆ Determine how clients will access the appliance. You can set up transparent or nontransparent client access to the appliance.
- ◆ Determine the type or types of protocols you will configure.

Information you need to gather

Before you begin configuring the appliance, gather the information described in the following table, based on the deployment decisions you have made.

Before deploying the appliance...	Do the following...
As an accelerator for one or more Web or RTSP streaming servers	Gather the following information about the accelerator: ◆ Host name of the accelerator (and virtual host names, if any) ◆ IP addresses of network interfaces ◆ Additional IP addresses to be used as IP aliases for the appliance to accelerate the servers ◆ Port or ports on which the appliance should accept requests from clients (for example, for HTTP requests, 80 and 3128) Note_____ The sum of appliance IP addresses and IP aliases should be equal to the number of servers to be accelerated. _____ Gather the following information about each server to be accelerated: ◆ IP address ◆ Ports (usually port 80 for Web servers) ◆ Domain names

Before deploying the appliance...	Do the following...
As an accelerator for an MMS streaming server	Gather the following information: ◆ IP address of the server to be accelerated ◆ IP addresses of appliance network interfaces ◆ Additional IP addresses to be used as IP aliases for the appliance to accelerate the servers Note_____ The sum of appliance IP addresses and IP aliases should be equal to the number of servers to be accelerated. _____
Outside a firewall with Web servers inside the firewall	Configure the firewall to allow requests from the appliance to be sent through the firewall.
And securing it from unauthorized client access	If you want to secure the appliance so that it cannot be used to resolve requests from unauthorized clients or caches outside your organization, gather the following information: ◆ IP addresses of any appliances you want to have permission to use this appliance to resolve requests ◆ IP addresses of any clients you want to have permission to use this appliance to resolve requests

Client access to an accelerator

You can provide client access to an accelerator using the following two methods:

- ◆ Configuring DNS to support an accelerator
- ◆ Using transparent proxying

Client access through DNS: When you configure DNS to support client access to an accelerator, you configure the accelerator's IP addresses and IP aliases in DNS to allow the accelerator to receive requests for the servers it is accelerating. You can set up DNS in the following ways:

- ◆ The accelerator handles all requests for the servers it is accelerating.
- ◆ Requests are split between the accelerator and the servers it is accelerating.

Client access through transparent proxying: To set up client access through transparent proxying, you use an L4 or L7 switch or WCCP router to provide access to an accelerator.

To support accelerators, the switch or router is deployed close to the accelerators. For details, see the transparency chapter in the *Administration Guide*.

Planning for uninterrupted service

Planning for load balancing in the event of disk failure

In the event of an appliance disk failure (in an acceleration deployment), the appliance might not be able to keep up with requests and will proxy them to the servers it is accelerating. This can be problematic if your back-end servers cannot handle the increased load. If you are concerned that your back-end servers are not powerful enough to handle the load if an appliance disk fails, you can configure your load balancers to query the appliance for disk health and redirect traffic accordingly.

Querying for disk health

To query the appliance for disk health, send a request to the following URL:

`http://cache_hostname:3132/cache_manager/disk_status`

The appliance will send one of the following disk status messages in response to the preceding HTTP request:

- ◆ If the appliance is handling requests normally, it will issue the following status message:

```
HTTP/1.0 200 OK
```

```
X-Cache-Status: DISK OK
```

- ◆ If the appliance cannot keep up with incoming requests, it will send the following status message:

```
HTTP/1.0 500 ERROR
```

```
X-Cache-Status: DISK BUSY
```

Changing the disk busy threshold

The default value of the disk busy threshold is 10. The default value of 10 indicates that 10 percent of misses were proxied to back-end servers in the last five minutes because the disk was too busy. Use the `config.http.disk_busy_thr` option to change the disk busy threshold. For example:

```
set cofig.http.disk_busy_thr //  
20  
//
```

Section B: Overview of accelerator configuration

About this section

This section provides detailed instructions for setting up Web acceleration and streaming acceleration. In addition, this section discusses Web acceleration and streaming acceleration rules.

Topics in this section

This section provides information about the following topics:

- ◆ [“Acceleration setup procedure”](#) on page 114
- ◆ [“DNS configuration to support an accelerator”](#) on page 118
- ◆ [“Appliance Manager options for configuring acceleration”](#) on page 122
- ◆ [“Continuing the accelerator configuration”](#) on page 126

Acceleration setup procedure

Setting up the appliance as an accelerator

To set up Web or streaming acceleration, complete the following steps.

Task	Description
1	In the Appliance Manager, select Setup tab > Network > Interfaces.
2	Use the Primary IP Address option to set up your accelerator primary IP address. Network Appliance recommends that you reassign the IP address and host name of the Web or streaming server, and assign the accelerator the IP address and host name previously assigned to the Web or streaming server. This enables the appliance to receive the HTTP and streaming media requests intended for the origin server. Note— If the appliance is being set up as a streaming MMS accelerator, you must reassign the IP address of the streaming MMS server, and assign to the accelerator the IP address that was previously assigned to the streaming MMS server.
3	If you want your accelerator to provide acceleration support for multiple Web or streaming servers, configure multiple IP aliases. Use the Aliases option to configure a single interface on the accelerator with multiple IP aliases (in addition to the accelerator’s primary IP address).

Task	Description
4	Select Setup tab > System > Host Name to set up your accelerator host name. ◆ If you are setting up the appliance as a Web accelerator, rename the Web server and assign the accelerator the name that was previously assigned to the Web server. From the user perspective, communication occurs directly with the Web server. The URL that users enter is the same as the URL that previously belonged to the Web server. ◆ If you are setting up the appliance as a streaming RTSP accelerator, rename the streaming RTSP server and assign the accelerator the name that was previously assigned to the streaming server. Note_____ Streaming MMS servers are accessed only by IP address. Therefore, if the appliance is being set up for streaming MMS acceleration, the host name can be arbitrarily assigned.
5	If you support client access to the accelerator through DNS, configure your DNS servers as follows: ◆ Associate your current Web or streaming RTSP accelerator host name with its current IP address. ◆ Associate your current Web or streaming server host names with their current IP addresses or IP aliases. ◆ If you are providing acceleration support for multiple Web or streaming servers, define multiple virtual accelerator host names and associate them with the IP aliases that you set up in Task 2. Note_____ Streaming MMS servers are accessed only by IP address, not host name; therefore, no DNS configuration applies to the streaming MMS servers or to the streaming MMS accelerators. For more details on DNS configurations supporting acceleration, see “DNS configuration to support an accelerator” on page 118.

Task	Description
6	Use the following options to specify the ports through which this accelerator will receive requests. ◆ If you are setting up a Web accelerator, select Setup tab > HTTP > General and use the HTTP Proxy Ports option to specify the ports through which this accelerator will receive HTTP requests. Configure one proxy port setting to port 80 so that the Web accelerator, which has essentially become the Web server, can listen on the same port (80) for requests. ◆ If you are setting up an RTSP streaming media accelerator, select Setup tab > Streaming > RTSP and use the RTSP Port option to specify the ports through which this accelerator will receive requests. ◆ If you are setting up an MMS streaming media accelerator, no port setting option is available. The fixed port is 1755.
7	If you are setting up a Web accelerator: ◆ Select Setup tab > HTTP > Web Server Acceleration > Rules tab > New Rule. Specify the Web server or servers to accelerate, and assign the accelerator or virtual accelerator hosts on this appliance to accelerate them. For details, click the Help icon. ◆ Select Setup tab > HTTP > Web Server Acceleration > General tab and set up specific options as follows: ❖ Web Accelerator Enable—Enable the Web accelerator function. ❖ Web Server Accelerator ACL—Specify ACL rules to apply to this accelerator. For details, click the Help icon.

Task	Description
8	If you are setting up a streaming accelerator: ◆ Select Setup tab > Streaming > Streaming Server Acceleration > Rules tab > New Rule. Specify the streaming server or servers to accelerate, and assign the accelerator or virtual accelerator hosts on this appliance to accelerate them. For details, click the Help icon. ◆ Select Setup tab > Streaming > Streaming Server Acceleration > General tab and set up specific options as follows: ❖ Streaming Server Acceleration Enable—Enable the streaming accelerator function. ❖ Streaming Server Acceleration ACL—Specify ACL rules to apply to this accelerator. For details, click the Help icon.

DNS configuration to support an accelerator

About configuring IP addresses in DNS

When you set up Web or streaming server acceleration, you configure the accelerator’s IP addresses and IP aliases in DNS so that the appliance can receive requests for a Web or streaming server. However, whether you want the accelerator to handle all requests for the Web or streaming server determines whether you configure DNS to include the IP address of the Web or streaming server that the appliance is accelerating.

Note
For streaming MMS servers, you can map only one host name to one IP address; mapping of multiple host names to multiple IP addresses is not supported.

About configuring the server IP addresses in DNS

The following table shows how to configure DNS, depending on whether you want the accelerator to handle all requests.

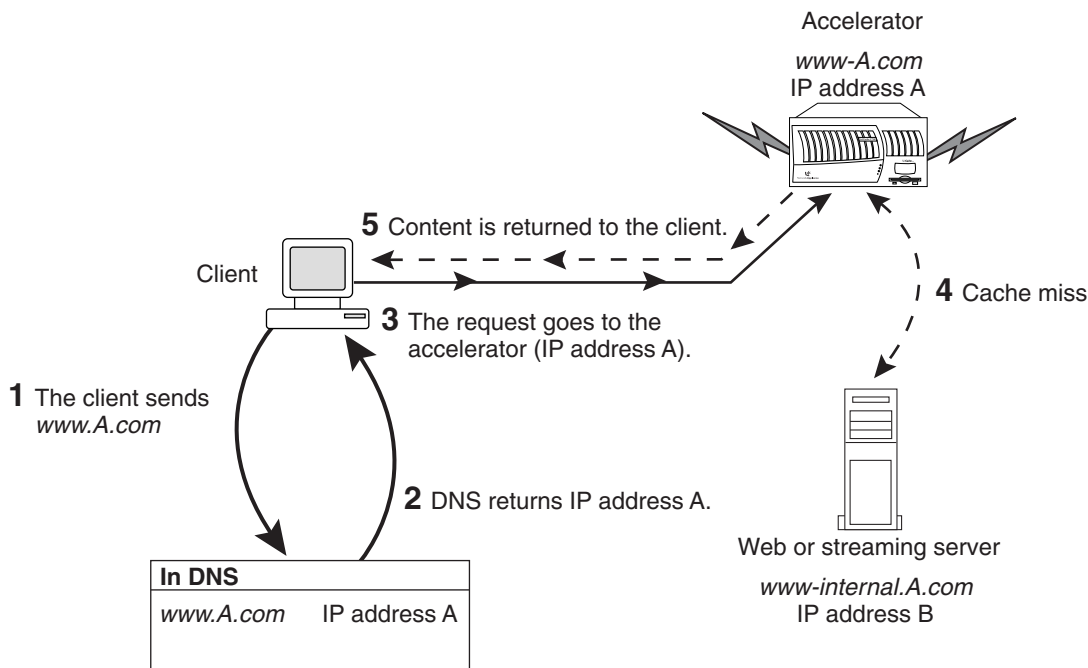
If you want...	Configure this in DNS...	Other requirements
The accelerator to handle all requests for the Web or streaming servers	Only the IP addresses of the accelerator, not those of the Web or streaming servers that the appliance is accelerating	The host name for the accelerator and the Web or streaming server must be different.
To split the requests between the accelerator and the Web and streaming servers	The accelerator IP addresses and the Web or streaming server IP addresses	None

Many organizations have the accelerator handle all the requests for the Web or streaming server. However, you might, for example, want to split requests between the accelerator and a Web or streaming server in either of the following circumstances:

- ◆ The link between the client and the Web or streaming server cannot handle the request load to the Web or streaming server.
- ◆ Neither the Web or streaming server nor the accelerator can handle the full request load.

Example of the accelerator handling all requests

The following illustration shows a deployment in which the accelerator handles all requests. To support this deployment, you configure DNS with the accelerator IP address but not the Web or streaming server IP address. Notice that in this type of deployment, the accelerator and the Web or streaming server have a different host name.



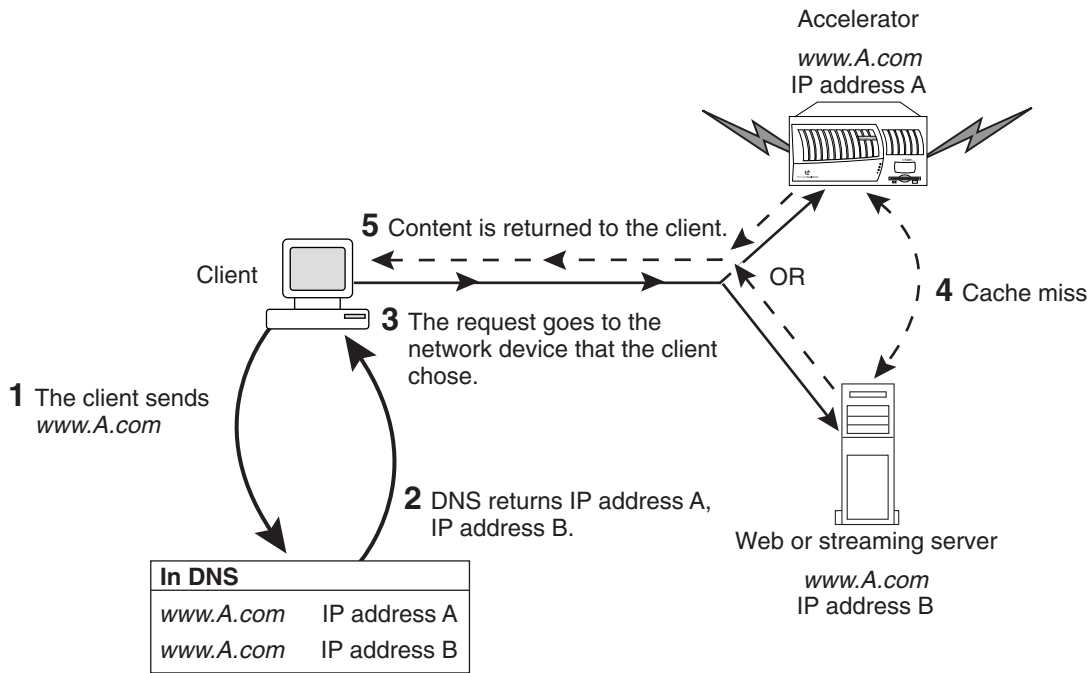
The flow of the request in the previous illustration is explained in the following table.

Stage	Description
1	◆ For Web acceleration, the client sends the URL <code>http://www.A.com</code>, which the user entered into the browser. ◆ For streaming acceleration, the client's media viewer is directed to send a streaming server request to <code>www.A.com</code>.
2	The DNS server resolves host name <code>www.A.com</code> as IP address A.
3	The client sends the request to IP address A, the accelerator.

Stage	Description
4	If the accelerator does not have the requested content in its cache (a cache miss), it fetches the content from the Web or streaming server.
5	The accelerator returns the requested content to the client; it returns the content immediately if it has the requested content in its cache.

Example of requests split between the accelerator and the server

The following illustration shows a deployment in which requests are split between the accelerator and the Web or streaming server. To support this deployment, you configure DNS with both the accelerator IP address and the Web or streaming server IP address. Notice that in this deployment, the accelerator and the Web or streaming server have the same host name.



The flow of the request in the previous illustration is explained in the following table.

Stage	Description
1	◆ For Web acceleration, the client sends the URL <code>http://www.A.com</code>, which the user entered into the browser. ◆ For streaming server acceleration, the client's media viewer is directed to send a streaming request to <code>www.A.com</code>.
2	The DNS server resolves host name <code>www.A.com</code> as IP address <code>A</code> .
3	The client sends the request to IP address <code>A</code> , the accelerator.
4	If the accelerator does not have the requested content in its cache (a cache miss), it fetches the content from the Web or streaming server.
5	The accelerator returns the requested content to the client; it returns the content immediately if it has the requested content in its cache.

Appliance Manager options for configuring acceleration

About Appliance Manager options

Options on the Appliance Manager Web Server Acceleration pages and the Streaming Server Acceleration pages enable you to carry out operations related to acceleration. These operations are described in the following sections.

About creating Web acceleration rules

When you create Web acceleration rules, you specify the Web server or servers to accelerate, and you assign the accelerator or virtual accelerator hosts on this appliance to accelerate them.

For example, the following rule maps HTTP requests directed at a specified IP address and host name on the accelerator to the DNS-listed host name and HTTP-defined host name on the Web server from which the requested object will be fetched. Accelerator access to a DNS lookup server is required.

Accelerator				Web server			
IP	Port	Host	Path	IP/DNS Host	Port	Host	Path
132.132.10.14	80	www.hplace.org	*	h_dns.org	8080	www.h_http.org	*

For information about creating and editing Web server acceleration rules, see the Appliance Manager online Help.

About ordering acceleration rules: When you create or edit a Web server acceleration rule, you can number that rule to specify the order in which it is applied in relation to other rules. In case of conflicting rules, the earlier-numbered rule is applied.

For information about creating and editing Web server acceleration rules, see the Appliance Manager Help.

About writing to rule specific logs: When you create or edit a Web server acceleration rule, you can specify a number ID for an optional log file which this accelerator generates especially for logging accelerator-server transactions generated by this rule.

If no value is specified, then the accelerator-server transactions generated by this rule are logged in the general Web access log.

If you have multiple rules for multiple Web servers, you can specify a different Web acceleration log number for each rule.

For information about creating and editing Web server acceleration rules, see the Appliance Manager Help.

About creating streaming acceleration rules

When you create streaming acceleration rules, you specify the streaming server or servers to accelerate, and you assign the accelerator or virtual accelerator hosts on this appliance to accelerate them.

RTSP acceleration example: The following rule directs the accelerator to fetch requested streaming objects through port 554 from the streaming server backplace.org in response to RTSP streaming requests directed to IP address 132.132.10.10, port 554, and host frontplace.org on the accelerator.

Accelerator				Streaming server			
IP	Port	Host	Path	IP/DNS Host	Port	Host	Path
132.132.10.10	554	frontplace.org	*	132.10.10.15	554	backplace.org	*

MMS acceleration example: The following rule directs the accelerator to fetch requested streaming objects from the streaming server at IP address 132.10.10.15, port 1755, when requests are directed to IP address 132.132.10.10 and port 1755 on the accelerator.

Accelerator				Streaming server			
IP	Port	Host	Path	IP/DNS Host	Port	Host	Path
132.132.10.10	1755	*	*	132.10.10.15	1755	*	*

For more information about creating and editing streaming server acceleration rules, see the Appliance Manager online Help.

MMS acceleration for HTTP-encapsulated data example: The following rule directs the accelerator to fetch requested streaming objects from the streaming server at IP address 132.10.10.15, port 1755, when requests are directed to IP address 132.132.10.10 and port 80 on the accelerator.

Accelerator				Streaming server			
IP	Port	Host	Path	IP/DNS Host	Port	Host	Path
132.132.10.10	80	*	*	132.10.10.15	1755	*	*

The appliance did not support server-side HTTP encapsulation in releases prior to NetCache 5.3.1. Therefore, to accelerate HTTP-encapsulated MMS data in earlier releases, you must construct acceleration rules as follows:

```
132.132.10.10 80 * * 132.10.10.15 80 * *
```

About regulating access through the Web or streaming accelerator

You can write ACL rules to accept or deny access to the requested content through the accelerator. For example, the following access control denies the client with IP address 128.126.10.0/24 from accessing the Web or streaming accelerator:

```
deny client-ip 128.126.10.0/24
```

For information on writing ACL rules, see the Appliance Manager online Help.

ContentDirector requirements for acceleration rules

If you are using ContentDirector to prefill content, you must use the URL of the origin server as the target URL of the ContentDirector push. Otherwise, your clients cannot access the prefilled content. The reason for this is that content must be cached under the URL of the origin server so that cache misses can be resolved from the origin server.

For example, the following acceleration rule correctly directs the Web accelerator to search for content that includes the URL of the origin server:

```
accelerator-ip 8080 * origin-server-ip 8080 *
```

Given the preceding rule, if you prefill (cache) content using the IP address of the Web accelerator, clients will not be able to access content. For example, consider content that is prefilled as follows:

```
http://accelerator-ip/foo.html
```

Because the acceleration mapping rules direct the appliance to look for content cached under the IP address of the origin server, your prefilled content will never be found.

About monitoring Web or streaming server status

You can configure the appliance to monitor for a transition in the status of Web servers or streaming servers that it is accelerating. When an accelerated server changes from an up state to a down state or from down to up, the appliance logs a syslog message and sends an SNMP trap if such a trap has been defined.

If a Web server is down, the appliance in monitor state will continue to serve freshly cached Web objects or return 404 (Not Found) codes for uncached or stale objects.

For details, see the Appliance Manager online Help.

About influencing when the appliance queries Web servers for modified objects

You can limit the IMS (If-Modified-Since) queries that the appliance sends to accelerated Web servers on cached Web objects through the IMS optimization option.

Select Setup tab > HTTP > Web Server Acceleration and locate the IMS optimization option. Click the Help icon for details.

Continuing the accelerator configuration

Optional features available The appliance includes a number of features you can use to customize and optimize the appliance for your environment. Use the following table to help you determine which features might be helpful to you.

Consider this feature...	To...	For more information, see...
Transparent redirect of client requests	Configure the appliance to accept client requests transparently redirected by a switch or router	Transparency chapter in the <i>Administration Guide</i>
IP takeover	Use IP takeover pairs to enable one appliance to take over services for another if one fails	Takeover chapter in the <i>Administration Guide</i>
Log file management and customization	◆ Enable or disable log files ◆ Select and modify log-file formats	Log file chapter in the <i>Administration Guide</i>

Section C: Accelerator configuration scenarios

About this section

This section provides several accelerator configuration scenarios. Each scenario identifies the configuration options that are used to support the scenario.

Topics in this section

This section provides information about the following topics:

- ◆ [“Scenario: Configuring an accelerator outside the firewall”](#) on page 128
- ◆ [“Scenario: Deploying an appliance as a distributed content site accelerator”](#) on page 131
- ◆ [“Scenario: Using multiple accelerators for a single Web or streaming server”](#) on page 134
- ◆ [“Scenario: Deploying one accelerator for multiple Web or streaming servers”](#) on page 138
- ◆ [“Scenario: Allowing limited access from another company”](#) on page 142

Scenario: Configuring an accelerator outside the firewall

About this scenario

In this scenario, the company wants a way to secure the Web content from intruders. It moved the Web or streaming server inside the firewall and put an appliance configured as an accelerator outside the firewall.

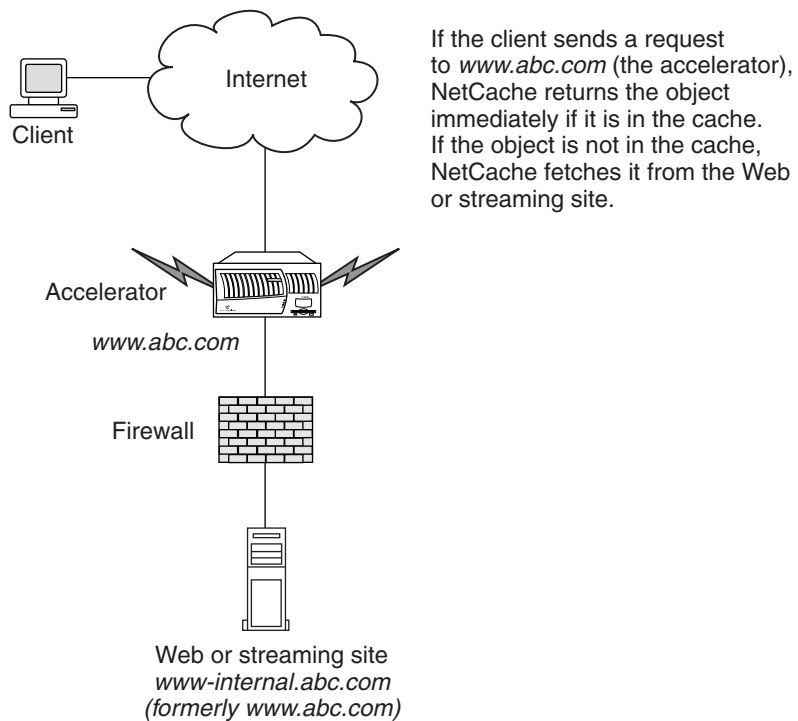
The locations of the Web or streaming server and accelerator are not apparent to users. The user who sends a request thinks that the request is being sent directly to the Web server or streaming server.

Appliance configuration outside a firewall

When you deploy an accelerator outside the firewall, you must configure the firewall so that it allows requests from the accelerator to pass through the firewall to the Web server or streaming server, and then back to the accelerator.

Illustration of the scenario topology

The following illustration shows a single accelerator outside the firewall.



Accelerator host name and IP address

Although the host name is not accelerator-specific, you might need to change it when you deploy the appliance as an accelerator.

The following table shows the configuration of the host name and IP address on the accelerator for this scenario.

Option	Configuration data	Comments
Host Name	www.abc.com	The accelerator takes the host name that was previously assigned to the Web or streaming server.
Primary IP Address	IP address A	—

After setup, you can change the Primary IP Address option (on the Network - Interfaces page) and the Host Name option (on the System - Host Name page) in the Appliance Manager.

Acceleration rule The following acceleration rule maps the accelerator IP address, port, and host name settings to the Web server that the appliance accelerates.

With this configuration, any client can access the accelerator because no restrictions are set for the IP addresses that can access it. No client has to directly access the Web server or streaming server, which is on the other side of the firewall.

Accelerator				Web server			
IP	Port	Host	Path	IP/DNS Host	Port	Host	Path
*	80	www.abc.com	*	www.internal.abc.com	8080	www.internal.abc.com	*

Scenario: Deploying an appliance as a distributed content site accelerator

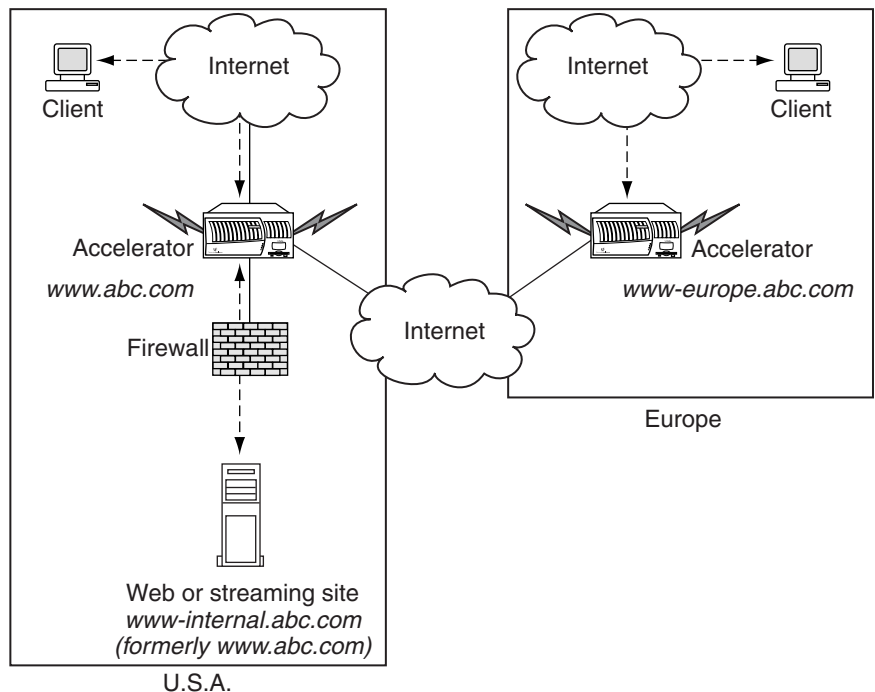
About this scenario

In this scenario, a large media site/hosting provider is interested in caching so that it can

- ◆ Constrain the cost of bandwidth
- ◆ Scale sites that are growing rapidly
- ◆ Secure the site content by putting the accelerator outside the firewall and the Web or streaming server inside the firewall

Illustration of the scenario topology

The following illustration shows this provider's deployment, which includes two appliances, one in the U.S.A. and one in Europe, running as accelerators.



About bandwidth savings with this deployment

An accelerator is deployed in Europe so that data from the Web site or streaming media site can be cached locally, thus saving expensive bandwidth, latency, or both. Initially, the European accelerator must obtain the Web objects or streaming media, thus incurring bandwidth costs. However, after the objects are in the cache in Europe, the objects can be served locally until they become stale. If the appliance determines that an object is stale, it fetches a newer version of the object from the Web server or streaming server.

U.S.A. and European accelerator configuration

The following table shows the accelerator options in the Appliance Manager that you would configure for each accelerator, and shows the host names of each accelerator. In this example, no restrictions were set on which IP addresses or netmasks could access the accelerators.

Option	Configuration data for the accelerator in the U.S.A.	Configuration settings for the accelerator in Europe
Host Name	www.abc.com	www-europe.abc.com
Primary IP Address	<i>IP address A</i>	<i>IP address Z</i>

After setup, you can change the Primary IP Address option (on the Network - Interfaces page) and host name option (on the System - Host Name page) in the Appliance Manager.

U.S.A. acceleration rule: The following acceleration rule is configured on the accelerator in the U.S.A.

Accelerator				Web server			
IP	Port	Host	Path	IP/DNS Host	Port	Host	Path
*	80	www.abc.com	*	*	8080	www.internal.abc.com	*

European acceleration rule: The following acceleration rule is configured on the accelerator in Europe.

Accelerator				Web server			
IP	Port	Host	Path	IP/DNS Host	Port	Host	Path
*	80	www.europe.abc.com	*	www.abc.com	80	www..abc.com	*

In the preceding rule on the European accelerator, the U.S.A. accelerator, `www.abc.com` is specified. In this case, if the European accelerator does not have a requested object in its cache, it sends a request to the U.S.A. accelerator to obtain the object. This method is preferable because you do not have the cost of the slow firewall or the bottleneck that prompted the deployment of the U.S.A. accelerator.

Note that the U.S.A. accelerator, `www.abc.com`, is not a parent to the European accelerator. You simply point the European accelerator to the U.S.A. accelerator instead of pointing it to the actual Web server or streaming server. The European accelerator is not aware of the U.S.A. accelerator, just as a user would not be aware of its presence.

Getting requests to the appropriate accelerator

One of the goals for this scenario is to reduce bandwidth cost. Therefore, the administrator wants to make sure that the users are sending requests to the closest accelerator. Assume for this configuration example that the administrator will inform users what URL to enter. The administrator instructs users in Europe to enter `http://www-europe.abc.com/` and instructs users in the United States to enter `http://www.abc.com/`.

Note

There are a number of other ways to enable requests to reach the appropriate accelerator. For example, you could use DNS routing, or you could use a product such as 3DNS (from F5 Lab) or Distributed Director (from Cisco) that can distribute requests to the closest accelerator.

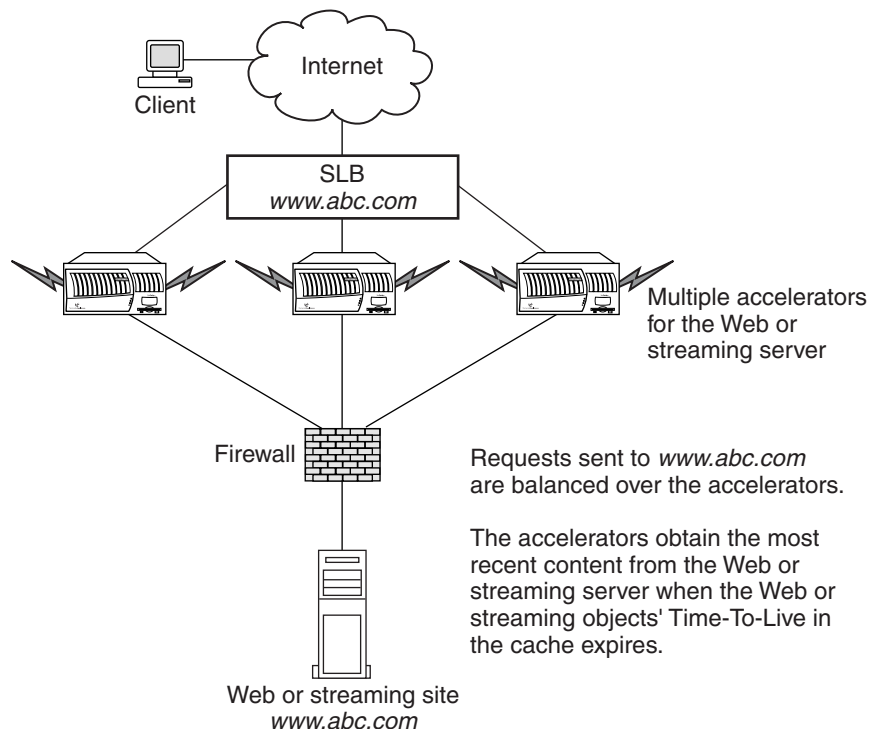
Scenario: Using multiple accelerators for a single Web or streaming server

About this scenario

In this scenario, the Web or streaming site is very big and very busy. One Web server cannot handle the amount of traffic. Instead of deploying multiple Web or streaming servers, multiple accelerators were deployed to reduce the load on the Web or streaming server.

Illustration of the scenario topology

The following illustration shows a single Web server or streaming server supported by multiple accelerators.



Using an SLB for request distribution

In this scenario, a Server Load Balancer (SLB) is deployed in front of the accelerators to provide a way to distribute incoming requests over the accelerators.

The SLB takes the name of the Web or streaming site (in this case, www.abc.com) instead of the accelerators taking the name of the Web or streaming site, as in the previous scenarios. When you use an SLB, you do not need to enable transparent proxying. The reason is that the host name in the URL that users enter to reach the Web server site is actually resolved as the IP address of the SLB.

Note

You could also use transparent proxying for distributing requests over the accelerators by deploying an L4 or L7 switch or a WCCP router in front of the accelerators.

Host name and IP address of the three accelerators

The following table shows the configuration of the host name and IP address for each of the three accelerators.

Data to provide	Configuration data for each accelerator	Comments
First accelerator’s host name and IP address	www-accel1-abc.com <i>IP address 1</i>	You can assign any host name that you want to each accelerator. For each accelerator, enter its unique IP address.
Second accelerator’s host name and IP address	www-accel2-abc.com <i>IP address 2</i>	
Third accelerator’s host name and IP address	www-accel3-abc.com <i>IP address 3</i>	

After setup, you can change the Primary IP Address option (on the Setup tab > Network > Interfaces page) and Host Name option (on the Setup tab > System > Host Name page) in the Appliance Manager.

Rules on each accelerator

The following acceleration rules are configured on the three accelerators.

Acceleration rule on www-accel1-abc.com:

Accelerator				Web server			
IP	Port	Host	Path	IP/DNS Host	Port	Host	Path
*	80	www-accel1-abc.com	*	www.abc.com	*	www.abc.com	*

Acceleration rule on www-accel2-abc.com:

Accelerator				Web server			
IP	Port	Host	Path	IP/DNS Host	Port	Host	Path
*	80	www-accel2-abc.com	*	www.abc.com	*	www.abc.com	*

Acceleration rule on www-accel3-abc.com:

Accelerator				Web server			
IP	Port	Host	Path	IP/DNS Host	Port	Host	Path
*	80	www-accel3-abc.com	*	www.abc.com	*	www.abc.com	*

Configuring the SLB

You must configure the SLB so that it can distribute the requests among the accelerators. How you configure an SLB to know about the accelerators varies according to manufacturer.

The following table shows the data that you need to configure on the SLB in this scenario.

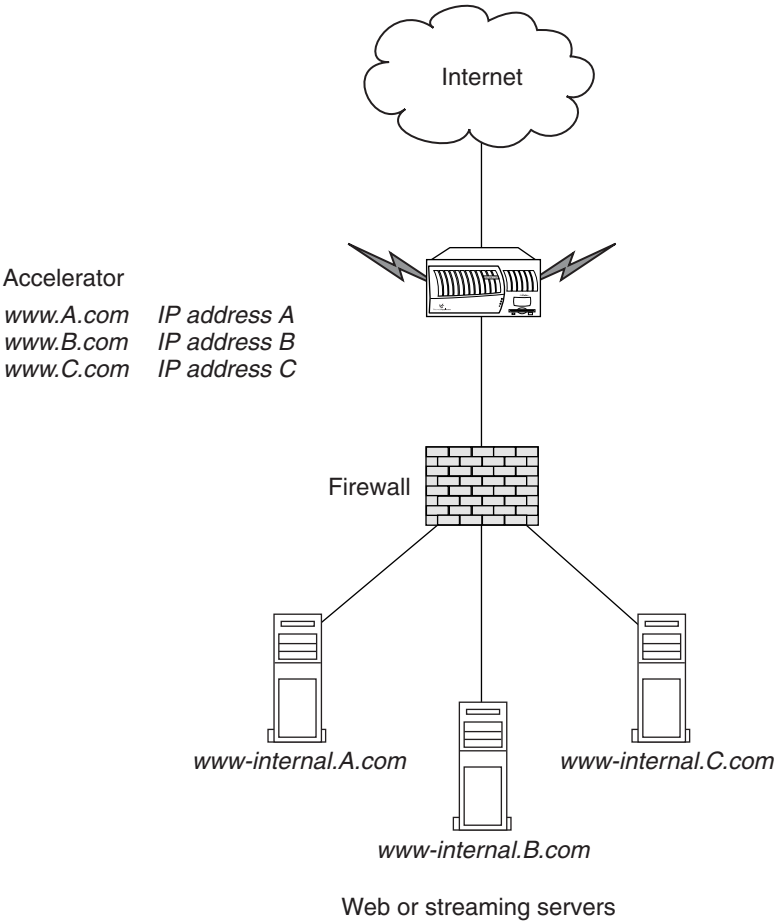
Data to provide	Configuration data	Comments
Host name	www.abc.com	You must assign the SLB the same host name as the Web server. Assigning the same host name to both devices will not cause a problem because they have different IP addresses. You make this host name known as the host name of the Web server.

Data to provide	Configuration data	Comments
IP address	<i>IP address P</i>	This is the virtual IP address of the SLB, that is, the IP address that you make known to the world as the IP address for the Web server. However, it is actually the IP address of the SLB.
How to distribute requests	Distribute based on load	The best approach is to configure the SLB to distribute requests based on load. It is likely that copies of the same object will be cached in multiple accelerators. However, this duplication is not a problem when deploying accelerators because the reason for deploying accelerators is to move data closer to users and to serve it quickly. In this scenario, the Web server's data is served quickly from three accelerators.

Scenario: Deploying one accelerator for multiple Web or streaming servers

About this scenario In this scenario, one accelerator is deployed for three Web servers or streaming servers.

Illustration of the scenario topology The following illustration shows a single accelerator servicing multiple Web servers or streaming servers.



Overview of configuration steps

- When you configure an accelerator to accelerate multiple Web servers or streaming servers, you take the following steps:
- ◆ Configure the host name and IP address for the accelerator.
 - ◆ Configure IP address aliases and host name aliases for the interfaces on the accelerator. You might, for example, want to have one interface handle requests for one Web server or streaming server.
 - ◆ Configure the accelerator options and rules, which enable the accelerator to send cache misses to the correct Web server or streaming server.

Accelerator host names and IP addresses

Assume for this scenario that only one network card is installed in the accelerator but that there are multiple IP addresses associated with that card. Therefore, the accelerator needs IP address aliases assigned to the interfaces.

The following table shows the configuration of the host name, IP address, IP address aliases, and virtual host names on the accelerator.

Data to provide	Configuration data	Comments
Host name*	www.A.com*	In this example, the host name that was assigned to the Web servers or streaming server becomes the host name of the accelerator. The original Web server or streaming server is renamed. You can assign any names to the Web servers or streaming servers. However, Network Appliance recommends that you do not assign the same host name to both the accelerator and the Web server or streaming server because this might be a problem for DNS. Note— Because a streaming MMS server is accessed by IP address only, the host name is not a critical setting on accelerators that service streaming MMS servers.
IP address*	IP address A	This is the IP address for the accelerator. When you need multiple IP addresses for the same interface, you need to assign IP address aliases. Doing so enables the accelerator to handle the requests for the multiple Web servers or streaming servers appropriately.

Data to provide	Configuration data	Comments
IP aliases (for IP addresses)	<i>IP address B</i> <i>IP address C</i>	In this example, a separate IP address alias is assigned to each interface on the accelerator. Use the IP Aliases option on the Network - Interfaces page to configure each IP address alias.
Virtual accelerator host names	<i>www.B.com</i> <i>www.C.com</i>	In this example, the IP alias addresses will handle requests for the second and third Web servers or streaming servers. A virtual accelerator host name associated with each IP alias must be assigned through DNS so that a particular IP alias address can handle requests for a particular Web server or streaming server. In this example, the Web servers or streaming servers are renamed, and the original names are used as the virtual accelerator host names. Note — Because a streaming MMS server is accessed by IP address only, the virtual accelerator host name configured for the accelerator is not required to support streaming MMS acceleration.

If necessary, you can change the IP address (on the Network - Interfaces page) and the host name (on the System - Host Name page) in the Appliance Manager.

Acceleration rules for multiple servers

The following acceleration rules map requests to specific Web or streaming servers.

Accelerator				Web Server			
IP	Port	Host	Path	IP/DNS Host	Port	Host	Path
<i>IP_addr_A</i>	*	www.A.com	*	www.internal.A.com	8080	*	*
<i>IP_addr_B</i>	*	www.B.com	*	www.internal.B.com	8080	*	*

Accelerator				Web Server			
IP	Port	Host	Path	IP/DNS Host	Port	Host	Path
<i>IP_addr_c</i>	*	www.C.com	*	www.internal.C.com	8080	*	*

The entries in these options map each Web server or streaming server to the accelerator or to a virtual accelerator host configured for this appliance.

The Web server’s or streaming server’s host name was www.A.com, but it was changed so that the accelerator could be assigned the host name.

How the appliance processes Web or streaming media requests

When a request is received by an accelerator, the accelerator first examines the IP address on which the request came in. Next, the accelerator examines the host name in the MIME header. The accelerator looks for a match of both the IP address and destination host name to determine the Web server or streaming server for which the request is intended. If the accelerator cannot match both the IP address and host name, it chooses the best possibility according to the IP address.

Scenario: Allowing limited access from another company

About this scenario

In this scenario, your company and another company are partners. Your appliance is running as an accelerator. Your domain accelerator appears to the appliance in the partner's domain as if it were a Web server or streaming server.

Goals for this scenario

You have the following goals:

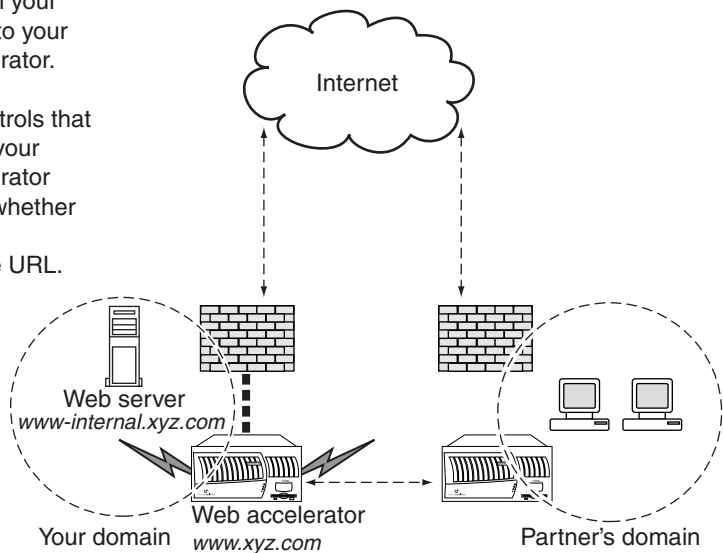
- ◆ You want certain clients in the partner company to be able to access only a predetermined set of URLs on specified Web servers in your domain.
- ◆ You want to be able to set different access controls for your company and your partner's company.

Illustration of the scenario topology

The following illustration shows the deployment for this scenario.

Requests from the partner's domain to URLs within your domain go to your Web accelerator.

Access controls that you set on your Web accelerator determine whether the request reaches the URL.



**Your accelerator
host names and IP
addresses**

The following table shows the configuration of the host name and IP address of the accelerator in your domain.

Option	Configuration data
Appliance Host Name	<code>www.xyz.com</code>
Primary IP Address	<code>IP_address_A</code>

After setup, you can change the Primary IP Address option (on the Network - Interfaces page) and the Appliance Host Name option (on the System - Host Name page) in the Appliance Manager.

**Network and URLs
to be allowed
access**

The IP address of the partner’s network for which you want to allow access is 128.125.51.0.

You want to allow clients on your partner’s network 128.125.51.0 access only to the following URLs on your company’s Web servers or streaming servers:

- ◆ `www.xyz.com/specs`
- ◆ `www.xyz.com/docs`
- ◆ `www.xyz.com/schedule`

**Acceleration rule on
your accelerator**

The following acceleration rule is configured on the accelerator in your domain.

Accelerator				Web server			
IP	Port	Host	Path	IP/DNS Host	Port	Host	Path
<code>*</code>	<code>80</code>	<code>www.xyz.com</code>	<code>*</code>	<code>www.-internal.xyz.com</code>	<code>8080</code>	<code>*</code>	<code>*</code>

**Accelerator ACL
rules**

The following ACL rules are configured on the accelerator in your domain.

```
allow client-ip 128.125.51.0 and (url-prefix
http://www.xyz.com/specs or http://www.xyz.com/docs or
http://www.xyz.com/schedule)
deny any
```

The appliance reads the settings in the Accelerator Access Control option as follows. If the request is from the specified network *and* the client from that network is trying to access one of the three specified URLs, then allow access. Deny all requests that do not match the rules.

- ◆ `client-ip 128.125.51.0` is the subnetwork from which clients in the partner's company can access your accelerator.
- ◆ `url-prefix http://www.xyz.com/specs` or `http://www.xyz.com/docs` or `http://www.xyz.com/schedule` are the URLs that clients on the partner's network can access.
- ◆ `deny any` specifies that no other networks can access the accelerator, and clients on the specified network can only access the three specified URLs. The `deny any` rule was entered at the end of the Accelerator Access Control option to allow only hosts on subnet 128.125.51.0 to access the accelerator.

Setting access controls for your company

You want to be able to set access controls for your company that are different from those you set for the partner's company. You can set the rules for your company on the Access Control Lists page in the Appliance Manager.

Because the appliance checks the access control rules in a specified order and because your accelerator access control specifies the subnetwork and URLs to be allowed for the partner's company, the access controls for your company and those for the partner company do not conflict.

Caution

Be sure that when you set up your access control options, you are not compromising your firewall security by allowing access from other outside companies.

About this chapter	This chapter provides information to help you plan to configure a NetCache appliance as a news cache. It includes a scenario in which NetCache appliances are deployed as news caches in an ISP network.
Before reading further	This chapter assumes that you have read the information about news caching in the <i>Deployment Guide</i> and have made decisions about how you want to deploy your NetCache appliance as a news cache on your network.
Sections in this chapter	This chapter contains the following sections: ◆ Section A, “Planning for news cache configuration,” on page 146 ◆ Section B, “How NetCache manages news objects,” on page 151 ◆ Section C, “Configuration options for a news cache,” on page 157 ◆ Section D, “News cache performance,” on page 162 ◆ Section E, “Monitoring a news cache,” on page 166 ◆ Section F, “News cache configuration scenario,” on page 171
Procedures and option descriptions	See the Appliance Manager online Help for configuration procedures and option descriptions.
Location of configuration options	To access the news caching options in the Appliance Manager, select: Setup tab > NNTP

Section A: Planning for news cache configuration

About this section

This section provides information to help you plan how to configure your NetCache appliance as a news cache.

Topics in this section

This section provides information about the following topics:

- ◆ [“Planning for network deployment configuration”](#) on page 147
- ◆ [“Determining the configuration requirements”](#) on page 148
- ◆ [“Additional features to consider implementing”](#) on page 149

Accessing the Appliance Manager and the CLI

After you have planned how to configure your NetCache appliance as a news cache, you can configure the appliance using the Appliance Manager (the NetCache browser-based administration interface) or the NetCache command line interface (CLI). For information about accessing the Appliance Manager, see the *NetCache Quick Setup* document. For information about the NetCache CLI, see the *Command Line Reference*.

Planning for network deployment configuration

Decisions to be made

While you read the *Deployment Guide*, you made decisions about deploying your NetCache appliance as a news cache on your network. Some of the deployment decisions you made require configuration using the Appliance Manager or the NetCache CLI.

The following table lists deployment decisions that require configuration, and it provides the location of information to help you plan how to configure the features that you want to implement.

If you want to...	See...
Deploy the appliance as a transparent cache with an L4 or L7 switch or a WCCP 2.0-based router	The transparent proxying chapter in the <i>Administration Guide</i>
Deploy the appliance with a firewall or with no firewall	The hierarchy chapter in the <i>Administration Guide</i>
Implement a load balancing strategy across multiple news caches	The transparent proxying chapter in the <i>Administration Guide</i>
Use IP takeover pairs to enable one appliance to take over services for another if one fails	The takeover chapter in the <i>Administration Guide</i>

Determining the configuration requirements

Decisions to be made

After you have planned for the network deployment configuration of your NetCache appliance, you should decide how you want to configure your appliance to function as a news cache on your network.

While you read this chapter, you will determine whether you want to do the following:

- ◆ Allow article posting through the news cache
- ◆ Identify a news server or another news cache to resolve cache misses for this news cache
- ◆ Apply access controls explicitly to authenticated users or to individuals as part of a group
- ◆ Impose authentication requirements to service NNTP requests through this news cache
- ◆ Configure the appliance to generate a trace header each time a client uses the news cache to post an article, which enables the news server to track spammers
- ◆ Specify the number of seconds to elapse before NetCache checks the host to determine whether a cached article was canceled
- ◆ Specify the number of seconds to elapse before NetCache checks the host for new and deleted articles
- ◆ Include a welcome message in the connect banner that is sent to clients when users log in to the news cache service
- ◆ Select the format in which log files are written

NNTP license requirement

Before you configure the NetCache appliance as a news cache, ensure that you install a valid NNTP license on the appliance. If you do not have an NNTP license, contact your Network Appliance sales representative or the Installed Base Group (ibg@netapp.com).

Additional features to consider implementing

Optional features available

NetCache includes a number of features that you can use to customize the NetCache appliance for your environment. Use the following table to help you determine which features to implement.

If you want to...	See...
◆ Customize the configuration of network interfaces ◆ Create virtual interfaces	The network interface chapter in the <i>Administration Guide</i>
Configure the appliance to use default and static routes for incoming and outgoing requests	The routing chapter in the <i>Administration Guide</i>
Configure the appliance for host name resolution and for DNS caching	The DNS chapter in the <i>Administration Guide</i>
Restrict administrative access for security purposes	The administrative security chapter in the <i>Security Guide</i>
Require users to authenticate themselves before NetCache services requests	The authentication chapter in the <i>Security Guide</i>
◆ Prevent unauthorized clients from accessing NetCache services ◆ Restrict the access of authorized clients to specified content and NetCache services	The access control chapter in the <i>Security Guide</i>
Establish direct tunneling through the appliance for traffic that the appliance does not handle	The protocol tunneling chapter in the <i>Administration Guide</i>
Configure the SNMP agent on the NetCache appliance to use diagnostic and other management services	The SNMP chapter in the <i>Administration Guide</i>
Enable or disable log files	The log files chapter in the <i>Administration Guide</i>
Select and modify log file formats	The <i>Guide to Messages and Log File Entries</i>
◆ Customize error messages ◆ Hide error messages from users	The error handling chapter in the <i>Administration Guide</i>

If you want to...	See...
Control bandwidth use	The bandwidth allocation topic in the Appliance Manager online Help
Configure the ContentDirector software to distribute content from a central source to remote NetCache appliances	The ContentDirector agent topic in the Appliance Manager online Help
Enable your NetCache appliance to interact with an ICAP server to adapt content to local policies	The <i>ICAP Services Guide</i>
Modify your Autosupport configuration settings	The Autosupport chapter in the <i>Administration Guide</i>

Section B: How NetCache manages news objects

About this section This section describes which news objects are cached and explains how NetCache interacts with news objects and news servers.

Topics in this section This section provides information about the following topics:

- ◆ “[Which news objects NetCache caches](#)” on page 152
- ◆ “[When NetCache checks for new articles](#)” on page 153
- ◆ “[When NetCache checks for canceled articles](#)” on page 155

Which news objects NetCache caches

Types of news objects cached

NetCache running as a news cache caches the following:

- ◆ News OverView (NOV) data for an article
Overview data is header information about the article that includes the message ID (the unique identifier for the article), date, and subject.
- ◆ Group list
The group list is a list of available newsgroups. Each line in the list includes the name of the newsgroup, the first and last known article in the newsgroup, and a flag indicating whether the server allows posting articles to the newsgroup.
- ◆ Group information
The group information consists of the numbers of the first and last articles in the group and an estimate of the number of articles in the group.

Caution

Group information is cached in volatile memory. Group information is lost if the NetCache appliance is rebooted.

- ◆ News articles
NetCache caches the entire article that the client requested, that is, the header and the body. The article header usually contains more detailed information about the article than is available from the News OverView data.

Types of news objects not cached

NetCache proxies commands for which it receives only partial data, for example, NEWNEWS, and sends them to the server. NetCache does not cache commands sent by the news server.

What you can change

You cannot change which news objects NetCache caches. However, you can modify how often NetCache checks with the news server to determine whether new articles have been posted, and you can modify how often NetCache checks to see if an article in its cache has been canceled. This enables you to determine how consistent you want NetCache to be with the news server.

The following sections provide information about changing these settings.

When NetCache checks for new articles

About checking for new articles

NetCache uses the cached group information to determine when to check the news server for new articles in a newsgroup.

When NetCache receives a request for a newsgroup, it first determines whether it has any information about the newsgroup. If NetCache has recent information about the newsgroup, NetCache provides that information to the client. Then, when the client requests an article in the newsgroup, NetCache provides the article from its cache or, if the object is not in the cache, retrieves the article from the news server.

However, if NetCache does not have recent information about the newsgroup, NetCache queries the news server to obtain this information. The frequency with which NetCache queries the news server for recent information about a newsgroup is set in the NNTP Refresh Rate option in the Appliance Manager (Setup tab > NNTP > Caching Behavior).

When NetCache checks for new articles

NetCache checks for new articles on the news server only when the following conditions are met:

- ◆ NetCache has had newsgroup information in its cache for a time period equal to or greater than the NNTP Refresh Rate setting (the default setting is 10 minutes).
- ◆ A NetCache client requests information about that newsgroup.

NetCache does not proactively check the news server for every newsgroup to determine whether new articles have been posted. Rather, NetCache waits until a client requests the newsgroup, then determines whether it has current information about that newsgroup. If it has current information about the news group, NetCache serves the information to the client. If it does not have current information about the news group, NetCache queries the news server for the information, caches a copy of the reply, and sends the information to the client.

How often NetCache should check for new articles

How often you want NetCache to check the news server for new articles depends on how consistent you want the NetCache information to be with the news server. The default NNTP Refresh Rate value is 10 minutes. This means that NetCache

will request information about new articles posted to the news server when it receives the first client request for information about that newsgroup after the 10-minute time period has expired.

For many organizations, the default value of 10 minutes provides a good balance between consistency with the news server and maximal hit rate and bandwidth savings. However, your organization might require a different setting in order to meet its goals.

Use the guidelines in the following table when changing the NNTP Refresh Rate setting.

If your organization’s priority is...	Then do this...
High degree of consistency between the information that NetCache has and the information on the news server	Set the NNTP Refresh Rate option to a very low value.
High hit rate and bandwidth savings	Set the NNTP Refresh Rate option to a high value.

The NNTP Refresh Rate option also directly impacts how NetCache updates the newsgroup list obtained from the NNTP LIST command. The list of newsgroups NetCache receives from the news server is a cacheable object. NetCache requests a new list of newsgroups if its object’s cached time has exceeded the value specified in the NNTP Refresh Rate option. This also means that users will see whether new newsgroups have been created or old newsgroups deleted no later than the setting in the NNTP Refresh Rate option.

When NetCache checks for canceled articles

What is a canceled article?

A canceled article is one that was posted to a news server but subsequently withdrawn by either the posting user or an administrator.

The posting user might cancel an article if, for example, the user made a mistake while creating the article. An administrator might cancel an article if, for example, the administrator is attempting to control abuses of the news service, such as advertisements posted to a newsgroup inappropriately.

About checking for canceled articles

NetCache uses the setting in the Canceled Article Time-To-Live option (Setup tab > NNTP > Caching Behavior) to determine when to check to determine whether an article has been canceled. The default is 3600 seconds (60 minutes). The Canceled Article Time-To-Live time period should be longer than the NNTP Refresh Rate (Setup tab > NNTP > General) time period.

When NetCache receives a request for an article, it first determines whether the article is in its cache. If the article is in the cache, NetCache then determines whether the article has been in the cache for a period equal to or greater than the Canceled Article Time-To-Live setting. If the article has been in the cache less than the Canceled Article Time-To-Live setting, NetCache serves the article to the client.

NetCache does not contact the news server (or other news cache) to find out about canceled articles until it receives a client request for an object whose Time-To-Live has expired.

If the article has been in the cache longer than the Canceled Article Time-To-Live setting, NetCache queries the host specified in the NNTP Server option to determine whether the article has been canceled. If the article has not been canceled, NetCache serves the article to the client from its cache. If the article has been canceled, NetCache sends a message to the client stating that the article has been canceled.

If this news cache points to another news cache, the Canceled Article Time-To-Live time period is the sum of the Canceled Article Time-To-Live time period that is set on this news cache and the Canceled Article Time-To-Live time period that is set on the other news cache.

**How often
NetCache should
check for canceled
articles**

How often you want NetCache to check the news server for canceled articles depends on how consistent you want the NetCache information to be with the news server. The default Canceled Article Time-To-Live value is 60 minutes. This means that NetCache will request information about an article when it receives the first client request for the article after the article has been in the cache for 60 minutes.

For many organizations, the default value of 3600 seconds (60 minutes) provides a good balance between consistency with the news server and maximal hit rate and bandwidth savings. However, your organization might require a different setting in order to meet its goals.

Use the guidelines in the following table when changing the Canceled Article Time-To-Live setting.

If your organization’s priority is...	Then do this...
High degree of consistency between the information that NetCache has and the information on the news server	Set the Canceled Article Time-To-Live option to a very low value.
High hit rate and bandwidth savings	Set the Canceled Article Time-To-Live option to a high value.

Section C: Configuration options for a news cache

About this section

This section describes all the configuration options, such as the port, the server IP address, authentication, and logging, available for your NetCache appliance news cache.

Topics in this section

This section provides information about the following topics:

- ◆ [“About the setup options”](#) on page 158
- ◆ [“About the logging options”](#) on page 161

About the setup options

Available setup options

You can set up your NetCache appliance as a news cache by configuring the following setup options in the Appliance Manager.

Options	Location
◆ NNTP License	Setup tab > System > Licenses
◆ NNTP Enable ◆ Article Posting Enable ◆ NNTP Port ◆ NNTP Server ◆ NNTP Welcome Banner ◆ NNTP ACL ◆ X-Trace-PostClient-IP Header Enable ◆ Authenticate NNTP Requests	Setup tab > NNTP > General
◆ Canceled Article Time-To-Live ◆ NNTP Refresh Rate	Setup tab > NNTP > Caching Behavior
◆ NNTP Access Log Enable ◆ NNTP Log Format	Setup tab > NNTP > Logging

NNTP License

Before you configure any of the news cache options, you must have installed a valid NNTP license on the NetCache appliance. If you do not have an NNTP license, contact your Network Appliance sales engineer or the Installed Base Group (ibg@netapp.com).

You can install the NNTP license during the initial setup or later, before you configure other news cache options.

NNTP Enable

This option enables or disables the NetCache appliance NNTP service.

Article Posting Enable

This option is a global setting that enables and disables the capability of clients to post articles through the news cache. However, the news server actually controls whether clients can post articles to particular newsgroups. Therefore, even if this option is enabled, the news server might prevent an individual client from posting articles.

If this option is disabled, no users using the news cache can post articles to the news server.

If this news cache points to another news cache, be sure to set this option the same way on both NetCache appliances. Otherwise, even if the news server allows posting, users of the news cache service will not be able to post articles.

NNTP Port

In this option, you enter the port number on which this NetCache appliance handles NNTP traffic. The default NNTP port number is 119. You can change the NNTP port number to any other number.

NNTP Server

In this option, you specify the IP address of the news server or another news cache that will resolve cache misses for this news cache. If you are entering the news server IP address and your news server components are running on different machines, enter the IP address of the Reader server.

You must also enter the port on the server or other news cache that listens for NNTP requests. Typically, the NNTP port is 119. If the server or other news cache is using a port other than 119 for NNTP, enter this port in this option.

NNTP Welcome Banner

In this option, you enter a text string, included in the connect banner, which is sent to a client and displayed when a user logs in to the news cache service. For example:

```
200 NetCache welcomes you to news - posting ok
```

The line begins with a result code and ends with an explanation of the result code. The result code is either 200 (posting ok) or 201 (posting not ok).

NNTP ACL

In this option, you can apply access controls explicitly to authenticated users or to individuals as part of a group. If this news cache points to another news cache (that is, if you have entered the IP address of another news cache in the NNTP Server option), configure access control only on this news cache (the “lower-level” cache).

X-Trace-PostClient-IP Header Enable

Enabling this option causes the header “X-Trace-PostClient-IP:” to be generated when a client that is running an NNTP session through this appliance posts an article.

The news server can use this header to track spammers.

Authenticate NNTP Requests

Enabling this option imposes authentication requirements to service NNTP requests through this NetCache appliance. For news caching, you can use any of the authentication mechanisms that NetCache supports—the NetCache user database, the LDAP server, the RADIUS server, or the NTLM authentication protocol. If this news cache points to another news cache (that is, if you have entered the IP address of another news cache in the NNTP Server option), configure authentication only on this news cache (the “lower-level” cache).

Canceled Article Time-To-Live

In this option, you specify the number of seconds that you want to elapse before, upon user request, NetCache checks the host specified in the NNTP Server option to determine if an article in the cache was canceled. If the news cache points to a higher level news cache, the setting of the Canceled Article Time-To-Live option is the total of the values set for this option on all of the news caches.

NNTP Refresh Rate

In this option, you enter the number of seconds to elapse before NetCache checks the host specified in the NNTP Server option for new and deleted articles in a newsgroup. If the news cache points to a higher level news cache, the NNTP Refresh Rate time period is the total of the values set for this option on all of the news caches.

Note

A new article is an article that has been posted recently on the news server and is not yet cached in the news cache. A deleted article is an article that has been deleted by the news server because it exceeded the server’s specified expiration time.

About the logging options

Available logging options

The following two logging options are available in the Appliance Manager:

- ◆ NNTP Access Log Enable
- ◆ NNTP Log Format

Location of the logging options: The logging options are available on the NNTP - Logging page (Setup tab > NNTP > Logging) in the Appliance Manager.

NNTP Access Log Enable

This option enables or disables the news cache log file.

The news cache provides a log file specifically for news caching. The file is called the NNTP access log.

NNTP Log Format

This option enables you to specify the format in which the log file will be written.

You can choose between NNTP Access Log Default Format and Custom Format.

NNTP Access Log Default Format: When you specify NNTP Access Log Default Format, the NNTP access log reports the following:

- ◆ Time, in seconds, since UNIX epoch, January 1, 1970
- ◆ Client IP address
- ◆ News article accessed by the client
- ◆ Newsgroup accessed by the client
- ◆ Total number of bytes sent by the NetCache appliance to the client
- ◆ Total number of bytes received by the NetCache appliance from the client
- ◆ Errors that occurred during transactions (NetCache appliance error codes)
- ◆ Errors that occurred during transactions (NNTP server error codes)
- ◆ NNTP transaction codes
- ◆ Total time taken to complete the transaction

Custom Format: The NetCache appliance custom log feature enables you to modify the format of the NNTP access log file so that you can tailor the logs to meet your requirements. You can create your own log file formats by selecting the Custom Format option on the NNTP - Logging page and entering the log headers directly into the Custom Format box.

Section D: News cache performance

About news cache tuning

A news cache does not accept or send news feeds. Therefore, a news cache administrator does not need to closely monitor disk usage, as is necessary for a news server. In addition, the administrator does not need to manage the news cache's interaction with news servers. After you configure your news caches, administration of the news caches is minimal.

When to use the NNTP access log

Use the NNTP access log if you want to view any of the following:

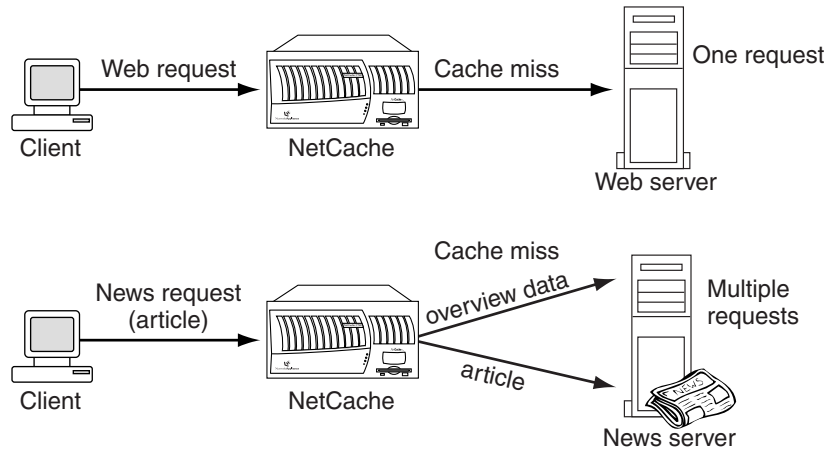
- ◆ Errors during transactions (NNTP codes)
- ◆ Number of groups and articles that a client accessed
- ◆ Time, in seconds and milliseconds, between a client connection and disconnection
- ◆ Number of bytes received from the client
- ◆ Number of bytes sent to the client
- ◆ Errors that NetCache generated
- ◆ Errors that the server generated
- ◆ Timestamp (in seconds and milliseconds)

Why the cache hit rate cannot be calculated for news caching

The *cache hit rate* is the number of user requests that NetCache resolves by using an object in the cache instead of fetching the object from the Web server. This definition is true for the cache hit rate for Web caching because a single Web request from a client remains a single request from the time that the client issues it to the time the client receives the requested data.

In contrast, one news request from the client can become multiple requests to the news server and, therefore, multiple responses from the news server. Therefore, the cache hit rate cannot be calculated for a news cache the same way as it can for a Web cache.

The following illustration compares how NetCache sends a Web request and a news request to the server.



NNTP bandwidth

NetCache calculates the percentage of bandwidth saved and displays it on the NNTP - General page under the Data tab of the Appliance Manager.

The fewer times that a news cache contacts a news server, the lower the bytes-from-server value.

NNTP statistics

The Appliance Manager displays the current status of NNTP request traffic and general statistics on cached and uncached articles served to clients. The statistics allow you to assess how well the news cache is functioning.

Consider the number of requests sent to the server on the NNTP - General page, and the request volume on the NNTP - Request Volume page under the Data tab.

Options that affect resolution time of news objects

The following configuration options on the NNTP - Caching Behavior page under the Setup tab in the Appliance Manager affect the response time for news requests, as well as the bandwidth savings:

- ◆ Canceled Article Time-To-Live option

This is the number of seconds that you want to elapse before, upon user request, NetCache checks the host specified in the NNTP Server option to determine if an article in the cache was canceled. The default value is 3600 seconds.

◆ NNTP Refresh Rate option

This is the number of seconds to elapse before NetCache checks the host specified in the Server IP Address and Port option for the arrival of new articles in a news group. The default value is 600 seconds (10 minutes).

The current status of NNTP request traffic and general statistics for cached and uncached articles returned to clients can be seen on the NNTP - General page under the Data tab in the Appliance Manager.

For details about the Canceled Article Time-To-Live and NNTP Refresh Rate options, see “[About the setup options](#)” on page 158.

Effect of the configured value on response time and bandwidth savings

For the Canceled Article Time-To-Live and NNTP Refresh Rate options, the higher the value you set, the less often NetCache contacts the news server. The less often NetCache has to check with the news server, the faster the response time and the better the bandwidth savings rate. However, some objects delivered to users from the cache might be stale.

Regardless of your settings of the Canceled News Article Time-To-Live and NNTP Refresh Rate options, NetCache verifies the consistency of an object whenever the period specified in either of these two options expires and the object is requested by a user.

Effect of adding an FQDN to the NNTP log

Adding a client’s fully qualified domain name (FQDN) by customizing (adding `c-dns` to the NNTP access log header) can result in reduced performance because of the additional time needed to perform DNS lookups. Increased latency also occurs when the FQDN lookup is added to other logs.

The NNTP access log header can be checked for the presence of the `c-dns lookup` on the NNTP - Logging page under the Setup tab in the Appliance Manager.

Consideration if the news cache points to another news cache

You can configure one news cache to point to another news cache rather than to a news server. For example, you might want to have news caches at POPs point to a news cache at a data center to reduce the number of connections to the news server. If one news cache is pointing to another, the NNTP Refresh Rate time period is the sum of the NNTP Refresh Rate values that are set on the first news cache and the other news cache that it points to. The same is true for the Canceled Article Time-To-Live option.

Increasing the probability of multiple requests for the same article

The average size of a news article is less than 4k, the block size of a NetCache appliance. However, the alt.binaries.* hierarchy routinely carries articles that exceed 100k. The alt.binaries.* hierarchy tends to include the most popular newsgroups at most sites.

If your cache contains many large objects, adding disk space to a system already fully populated with disks can increase the probability that articles in the cache will be requested by more than one user.

Section E: Monitoring a news cache

About this section

This section describes ways you can access NNTP-related data, which you can use to monitor your news cache.

Accessing monitoring information

You can monitor your news cache by accessing the following NNTP data from the Appliance Manager:

- ◆ NNTP statistics pages
- ◆ Access log files

Topics in this section

This section provides information about the following topics:

- ◆ [“NNTP statistics pages”](#) on page 167
- ◆ [“NNTP access log files”](#) on page 169

NNTP statistics pages

About NNTP statistics pages

The NNTP statistics pages report the current status of and general statistics about your news cache. The following Appliance Manager pages provide you with NNTP statistics information:

- ◆ NNTP - General
- ◆ NNTP - Data Volume
- ◆ NNTP - Request Volume

About the NNTP - General page

The NNTP - General page (Data tab > NNTP > General) displays statistics in the following two sections:

- ◆ Current Status
- ◆ General Statistics

Current Status: The Current Status section displays the number of active client and server connections, number of NNTP requests per second, and number of NNTP bytes read and written per second on the news cache. For detailed information about each field of the Current Status section, see the online Help available for each field.

General Statistics: The General Statistics section displays the cached and uncached articles sent to the clients, cached and uncached group information sent to the clients, post requests served, and requests sent to the server. For detailed information about each field of the General Statistics section, see the online Help available for each field.

About the NNTP - Data Volume page

The NNTP - Data Volume page (Data tab > NNTP > Data Volume) displays the bytes that the news cache sends to and receives from clients, and the bytes that the news cache sends to and receives from the servers. For detailed information about each field of the NNTP - Data Volume page, see the online Help available for each field.

**About the NNTP -
Request Volume
page**

The NNTP - Request Volume page (Data tab > NNTP > Request Volume) displays statistics about cacheable requests, local requests, and proxy requests. For detailed information about each field of the NNTP Statistics - Request Volume page, see the online Help available for each field.

NNTP access log files

About NNTP access log files

The Log Files - NNTP Access page (Data tab > Log Files > NNTP Access) enables you to access all available news cache log files. This page also enables you to copy, send through FTP and e-mail, and delete the access log files.

You can use the access log file to monitor the total number of bytes that the NetCache appliance sends to and receives from clients, errors that occurred during the transaction (NetCache appliance and NNTP error codes), NNTP transaction codes, and so on. For a complete list of options that you can monitor, see “[NNTP Log Format](#)” on page 161.

About the log file format

The NNTP access log file format depends on the option you selected while setting up the configuration options for logging. For detailed information about the log file format, see “[NNTP Log Format](#)” on page 161.

Accessing the NNTP access log file

To access the NNTP access log file, complete the following steps.

Step	Action
1	In the Appliance Manager, select Data tab > Log Files > NNTP Access.
2	In the “Select a Log File” list, select the log file that you want to access.

Step	Action
3	In the Log File Action option, select one of the following options: ◆ View the File Enables you to view the entire file. ◆ View the Last <i>x</i> Lines of the File Enables you to view the last <i>x</i> number of lines of the access log file that you selected in Step 2. The default number of lines is 50. ◆ Delete the File Enables you to delete the file that you selected in Step 2. ◆ Copy the File to Your Computer Enables you to copy the file that you selected in Step 2 locally to your computer. ◆ Copy the File to Your Computer in gzip Compressed Format Enables you to copy the file that you selected in Step 2 locally to your computer in gzip file-compression format. ◆ Send the File to the Following FTP or Web Server Enables you to send the file that you selected in Step 2 through FTP or HTTP to the location you specify in this option's text box. You can use the following format to specify a user name and password, and a port if required by the FTP or HTTP server: ❖ ftp://user:password@server:port/path/file ❖ http://user:password@server:port/path/file ❖ https://user:password@server:port/path/file
4	Click Submit at the bottom of the page.

Section F: News cache configuration scenario

About this scenario

In this news cache configuration scenario, an ISP provides both Web service and news service to its customers. The ISP has a data center with multiple NetCache appliances, each of which is dedicated to either Web caching or news caching. The news server is at the data center.

The ISP has a number of POPs (points of presence), each of which has a NetCache appliance. For simplicity, this scenario shows only two POPs. One POP has a dedicated news cache, and the other POP has a NetCache appliance that is both a news cache and a Web cache. The scenario demonstrates the configuration differences in both deployments.

What is included in the configuration examples

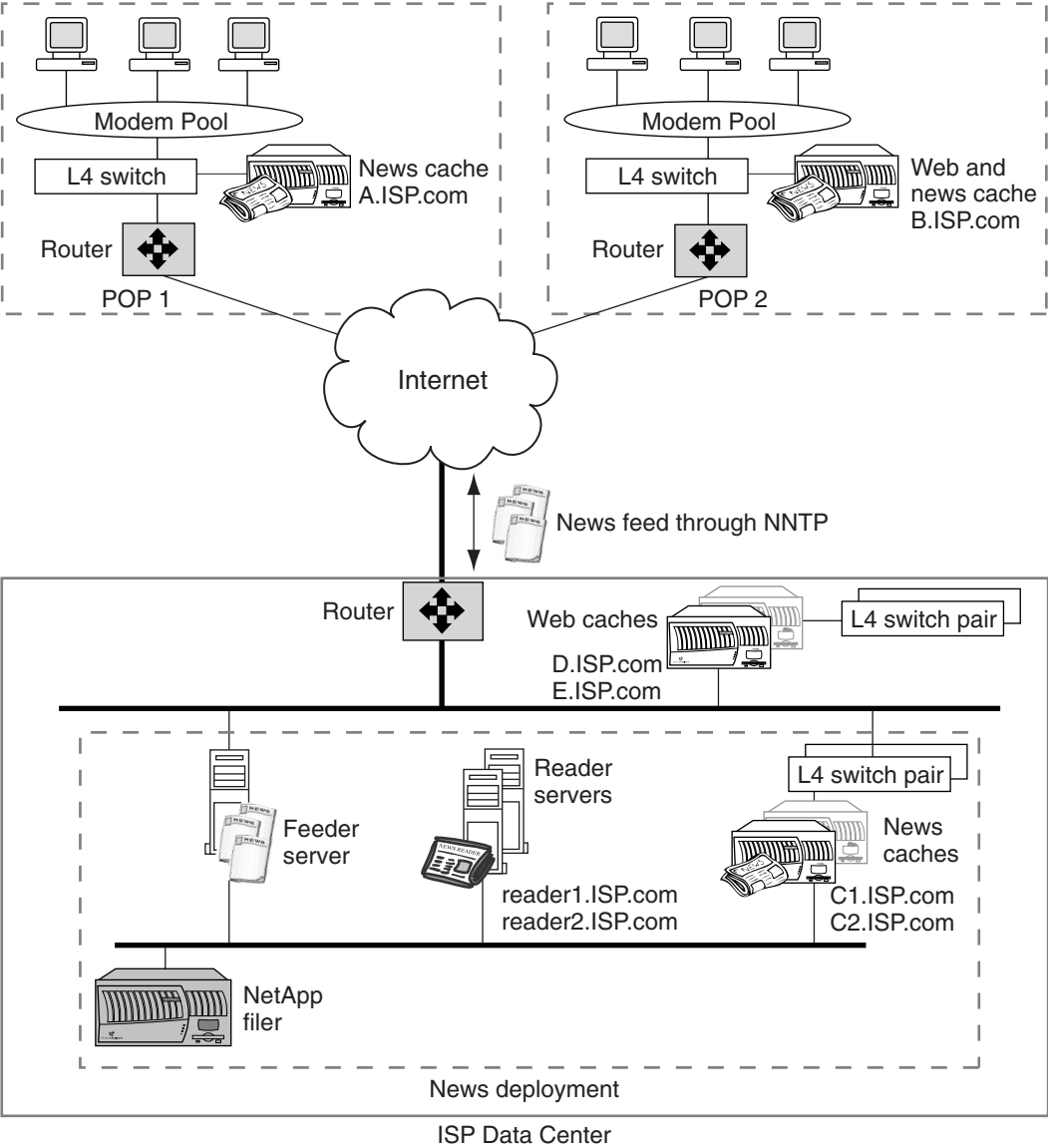
The configuration examples focus on the NNTP configuration options that would be set on each NetCache appliance. However, the example of the NetCache appliance that is both a news cache and a Web cache also includes options necessary to send Web request cache misses to the data center, and a few other options that you would use with Web caching but not with news caching.

Note

The configuration examples assume that you have completed the tasks required to identify the NetCache appliance to the network, such as specifying an IP address, a host name, and DNS.

Deployment illustrated

The following illustration shows the deployment of NetCache appliances at the POPs and data center.



Method chosen for client access to the news caches

The NetCache appliances at the POPs are configured for transparent proxying so that users do not have to configure their browsers to use the NetCache appliance service. At each POP, an L4 switch is deployed in front of the NetCache appliance to pass NNTP traffic (at POP 1) or HTTP and NNTP traffic (at POP 2) transparently to the appliance deployed at that POP.

The ISP advertises just one IP address for the news service. All users of this ISP are given the same host name to use for the news service. DNS is set up so that requests are routed to the closest POP. The reason for such a DNS configuration is that if users move, they can use the same host name for news even though they are dialing into a different POP.

- Failover strategy**
- The failover strategy for this scenario is as follows:
- ◆ Two news caches are deployed at the data center so that news service is still available if one news cache goes down. If both news caches go down, however, news service will be unavailable.
 - ◆ An L4 switch pair is deployed in failover mode in front of the news caches at the data center. If one switch fails, the other switch ensures connectivity to the news caches.
 - ◆ Two Web caches are deployed at the data center so that Web caching service is still available if one Web cache goes down. If both Web caches go down, the switch passes Web requests directly to the Internet.
 - ◆ An L4 switch pair is deployed in failover mode in front of the Web caches at the data center. If one switch fails, the other switch ensures connectivity to the Web caches.

Note

The switch is not configured to pass Web requests to the news cache and vice versa if all caches of one type go down. The reason is that the Web caches are not configured for news caching and the news caches are not configured for Web caching.

Configuration of the news cache at POP 1

The NetCache appliance at POP 1 is set up as a news cache, with caching behavior, logging, authentication, and transparency options configured. The following table shows the options that were configured for this NetCache appliance.

Option	Configuration data for the POP 1 NetCache appliance	Option location in the Appliance Manager
Basic setup of the news cache		
NNTP License	<nnnnnnnnnnnn>	Setup tab > System > Licenses
NNTP Enable	Enabled	Setup tab > NNTP > General
Article Posting Enable	Enabled	Setup tab > NNTP > General
NNTP Port	119 (default)	Setup tab > NNTP > General
NNTP Server (IP Address or Host Name)	<IP address for C1.ISP.com>	Setup tab > NNTP > General
NNTP Server (Port)	119 (default)	Setup tab > NNTP > General
NNTP Welcome Banner	Welcome to the Friendly ISP	Setup tab > NNTP > General
Caching behavior settings for the news cache		
Canceled Article Time-To-Live	3600 (Seconds)	Setup tab > NNTP > Caching Behavior
NNTP Refresh Rate	600 (Seconds)	Setup tab > NNTP > Caching Behavior
Log file settings for the news cache		
NNTP Access Log Enable	Enabled	Setup tab > NNTP > Logging
NNTP Log Format	NNTP Access Log Default Format	Setup tab > NNTP > Logging
Authentication settings for the news cache		
Authenticate NNTP Requests	Enabled	Setup tab > NNTP > General
Authenticate Client Requests Using These Protocols > Authenticate NNTP Requests	Enabled	Setup tab > Authentication > General
Authentication Checking Order	Change the default order to RADIUS and to None for all other lists	Setup tab > Authentication > General

Option	Configuration data for the POP 1 NetCache appliance	Option location in the Appliance Manager
RADIUS Enable	Enabled	Setup tab > Authentication > RADIUS
RADIUS Servers	Server information in <i>IP address:Port:password</i> format	Setup tab > Authentication > RADIUS
Transparency settings for the news cache		
Transparency > NNTP	Enabled	Setup tab > Transparency and IP Spoofing Enable

Configuration of the news cache and Web cache at POP 2

The NetCache appliance at POP 2 is set up as a Web cache and a news cache. For the Web cache, the transparency, clustering, access control, and Web filtering options are configured. For the news cache, the transparency, logging, and authentication options are configured. The options that were configured for the news cache setup are the same as those discussed in [“Configuration of the news cache at POP 1”](#) on page 173. The following table shows the options that were configured for the Web cache setup on this NetCache appliance.

Option	Configuration data for POP 1 NetCache appliance	Option location in the Appliance Manager
Transparency settings for the Web caches		
Transparency > HTTP	Enabled	Setup tab > Transparency and IP Spoofing Enable
Transparency > FTP	Enabled	Setup tab > Transparency and IP Spoofing Enable
HTTP Proxy Ports	80	Setup tab > HTTP > General
Hierarchy settings for the Web caches		
Bypass Hierarchy for Noncacheable Objects	Enabled	Setup tab > Hierarchies > General
Bypass Hierarchy for These Local Domains	Disabled	Setup tab > Hierarchies > General

Option	Configuration data for POP 1 NetCache appliance	Option location in the Appliance Manager
Hierarchies - Parents settings	D.ISP.com and E.ISP.com identified as parents. For detailed information about hierarchies, see the <i>Administration Guide</i>. Note_____ NetCache automatically creates a hierarchy forwarding rule with D.ISP.com and E.ISP.com as members of a cluster at the data center.	Setup tab > Hierarchies > Parents > New Parent
Access control settings for the Web caches		
HTTP ACL	Access denied to the following URL: deny url http://www.badsite.com	Setup tab > Access Control > Access Control Lists
SmartFilter Enable	Enabled	Setup tab > Access Control > SmartFilter > General
Select the Categories You Want to Block	Categories Sex and Hate Speech are blocked.	Setup tab > Access Control > SmartFilter > General

Additional features configured on the POP 2 NetCache appliance

The POP 2 NetCache appliance is both a Web cache and a news cache; therefore, some of the options configured for the POP 2 NetCache appliance are not shown for the POP 1 NetCache appliance. Some of the additional features are as follows:

- ◆ The NetCache appliance logical settings for hierarchies and clusters are not applicable to news caches. However, it is necessary to set up a cluster parent for the POP 2 NetCache appliance so that HTTP, FTP, Gopher, and tunnel requests that are cache misses at POP 2 can be sent to the Web caches at the data center.

- ◆ SmartFilter is enabled for Web caching at POP 2 to control access to URLs by category. SmartFilter is not applicable to news caching.

Configuration of news caches at the data center

At the data center, two NetCache appliances (C1.ISP.com and C2.ISP.com) are set up as news caches. Distributed caching cannot be set up on the news caches; therefore, failover and load balancing can be achieved by configuring the L4 switch pair that is in front of these NetCache appliances.

The following table shows the options that were configured for these NetCache appliances.

Option	Configuration data	Option location
Basic setup of the news cache		
NNTP License	<nnnnnnnnnnnn>	Setup tab > System > Licenses
NNTP Enable	Enabled	Setup tab > NNTP > General
Article Posting Enable	Enabled	Setup tab > NNTP > General
NNTP Port	119 (default)	Setup tab > NNTP > General
NNTP Server (IP Address or Host Name)	<IP address for reader1. ISP.com>	Setup tab > NNTP > General
NNTP Server (Port)	119 (default)	Setup tab > NNTP > General
NNTP Welcome Banner	Welcome to the Friendly ISP	Setup tab > NNTP > General
Caching behavior settings for the news caches		
Canceled Article Time-To-Live	3600 (Seconds)	Setup tab > NNTP > Caching Behavior
NNTP Refresh Rate	600 (Seconds)	Setup tab > NNTP > Caching Behavior
Log file settings for the news caches		
NNTP Access Log Enable	Enabled	Setup tab > NNTP > Logging
NNTP Log Format	NNTP Access Log Default Format	Setup tab > NNTP > Logging

Index

A

- access controls
 - configuring for Web cache performance 32
 - for Web or streaming accelerator 142, 144
- alt.binaries.* news hierarchy 165
- Apple QuickTime
 - NetCache support for 45
 - See also* QuickTime
- articles, news
 - average size of 165
- authentication 74
 - streaming media and 74

B

- bandwidth savings
 - calculating for NNTP 163
 - news caching 164
 - Web or streaming accelerator deployment 132
- byte hit rate
 - versus object hit rate 30
 - See also* URL byte hit rate

C

- cache control headers 13
- cache, news. *See* news caching
- cache, streaming media. *See* streaming media cache 70
- cache, Web. *See* Web cache 9
- caches 42
- Complete Fetch option
 - on a Web cache 17
 - use with Web accelerator 17
 - use with Web and news caching 17
- confill command 69
- connections
 - FTP 25
 - NetCache establishing 25
 - server establishing 25
 - streaming media 76, 79
 - HTTP for 102

- limiting client connections 75
- TCP 77
- UDP 77

- ContentDirector
 - restrictions 50, 82
 - use with streaming media 49
- Cookie Cheat Extensions
 - configuration option 30
 - described 30
 - values to add 31
- Cookie Cheat File Name Extensions
 - description of 11

F

- failover
 - examples for news caching 173
- firewalls
 - configuring for streaming media 87
 - examples 127
 - Web or streaming accelerator outside 128
 - for streaming media 91
 - effects of port blocking 48
 - ports to open 53, 54, 56
 - quality of streams 48
- Foundry switches, minimum firmware versions for streaming media 45
- FTP 22
 - connections 25
 - NetCache establishing 25
 - PASV 26
 - PORT 26
 - server establishing 25
 - consistency of FTP objects in cache 23
 - defined 22
 - NetCache support for 22
 - over HTTP 22
 - port 22
 - transparent 22
 - transparent client access to NetCache 22
 - transparent proxying, defined 22

H

- hierarchies, streaming media and 43
- hierarchy, news 165
- HTTP
 - and streaming media 47
 - authenticating Web users 18
 - streaming media and 92

I

- ICAP, streaming media and 43
- IP spoofing, streaming media and 55

L

- L4 switch
 - in front of accelerator 135
- L7 switch
 - in front of accelerator 135
- licenses
 - NNTP, how to obtain 158
 - streaming media 45
- Log FQDN option
 - disabling to improve performance 30
- logging, connection data for streaming media 71

M

- media players, supported 46
- media streams
 - client log forwarding for 79
- Microsoft Media Server. *See* MMS
- MMS
 - NetCache support for 45
- monitoring, streaming media 76
- multicast
 - back-channel, NetCache support for 55
- multicast streaming
 - features 58
 - IP address requirements 58
 - MMS multicasting 104
 - port requirements 58

N

- NetCache 29
- NetCache streaming accelerator. *See* streaming accelerator
- NetCache Web accelerator. *See* Web accelerator
- NetCache Web cache. *See* Web cache
- news articles, average size of 165
- news caching
 - average size of news article 165
 - Canceled Article Time-To-Live option 163
 - client access to news cache 173
 - configuration examples 171, 173
 - configuring NetCache for 145
 - enabling NetCache for 158
 - failover examples 173
 - how NetCache sends requests 162
 - NetCache pointing to another news cache 164
 - tuning response time for 162
 - See also* NNTP
- news objects
 - checking for canceled articles 155
 - checking for new articles 153
 - how NetCache manages objects 151
 - which objects are cached 152
- news requests, how sent 162
- NNTP
 - bandwidth savings
 - how calculated 163
 - canceled articles
 - checking for 155
 - definition of 155
 - configuration examples 171
 - enabling NetCache for 158
 - license, how to obtain 158
 - objects
 - checking for canceled articles 155
 - checking for new articles 153
 - port, description of 159
 - refresh rate 160
 - request resolution response time, NetCache
 - options affecting 163
 - server, description of 159
 - statistics 167
 - traffic, port to handle on 159
 - welcome banner 159

See also news caching

NNTP Refresh Rate option
relationship to response time 163

O

objects, news

- checking for new articles 153
- how NetCache manages 151
- which objects are cached 152

objects, Web

- caching objects with cookies 30
- calculating default TTL 13
- how long objects remain in the cache 12
- how long Web objects remain in the cache 14
- how to change refresh rate 16
- If-Modified-Since (IMS)
 - verifying object freshness 23
- making objects uncacheable 19
- specifying which objects are cached 8
- TTL Patterns option 14

P

PNA restrictions 87

port 119, used for news caching 159

port 21 (FTP) requests 22

ports

- configuring the proxy port 19
- FTP 22
- streaming media 96, 98, 100
 - recommended for transparency 53, 54, 56

prefilling

- commands for 69
- confill command 49, 59
- streaming media restrictions 50, 82

Progressive Networks Audio. *See* PNA

protocols

HTTP

- streaming media and 47

PNA 87

- supported for transparent proxying 22

Proxy port

- configuring 19
- Web cache settings 19

Q

QuickTime

- firewall configuration for 87

- NetCache support for 45

- UDP ports and 56

R

RDT 81

Real Data Transport. *See* RDT

Real Time Streaming Protocol. *See* RTSP

RealFlash 81

RealMedia 81

RealNetworks

- content types not supported by NetCache 81
- IP spoofing restrictions 88
- NetCache support for 45
- PNA 87
- RealProxy restrictions 55
- supported RealMedia MIME types 81

RealPix 81

RealSystem, NetCache support for 45

RealText 81

Real-Time Monitoring Protocol 72

Reload Rules 10

request resolution, FTP 22

requests

- how NetCache sends Web and news requests 162

response time

- for news requests 163
- for Web requests 163
- news caching 164

RTMON 72

RTSP, NetCache support for 45

S

Server Load Balancer, in front of Web or streaming accelerator 135

SLB. *See* Server Load Balancer

SmartFilter, with streaming media 42

SMIL 81

statistics, NNTP 167

streaming accelerator 144

- distributed 131
- DNS configuration for 118
- examples 128
 - handling all requests 119
 - limited access from another company 142
 - multiple accelerators 131
 - multiple servers and one accelerator 138
 - splitting requests with streaming server 120
- L4 switch in front of 135
- L7 switch in front of 135
- limited access from a partner 142
- outside the firewall 128
- Server Load Balancer in front of 135
- streaming media
 - acceleration 35
 - authentication
 - end-to-end 74
 - cacheability 81
 - caching
 - ejecting locked objects 70
 - how objects are stored 83
 - methodology 83
 - object expiration 84
 - object validation 84
 - RealNetworks content types and 81
 - restrictions 81
 - supported vendors 81
 - client access to cache, transparent proxying 53, 54, 56
 - client authentication 74
 - client monitoring for 72
 - client UDP requests 77
 - client-side HTTP encapsulation 102
 - configuration
 - planning 61
 - strategies 47
 - confill command 69
 - connections for 76, 79
 - TCP 77
 - UDP 77
 - content fill
 - command to use 69
 - example of 69
 - syntax 69
 - content providers 71
 - ContentDirector and 49, 50, 82
 - dynamic stream switching 79
 - firewalls
 - effects of port blocking 48
 - port blocking 91
 - ports to open 53, 54, 56
 - quality of streams 48
 - HTTP for 47
 - IP spoofing, restrictions 55, 88
 - limiting client connections 48, 75
 - logging connection data 71
 - MMS, defined 45
 - MMS, support for 45
 - monitoring information 76
 - multicast streaming
 - features 58
 - IP address requirements 58
 - MMS multicasting 104
 - port requirements 58
 - nontransparent redirection 65
 - objects, how stored 83
 - ports 96, 98, 100
 - prefilling
 - commands for 69
 - confill command 49, 59
 - restrictions 50
 - prefilling restrictions 82
 - protocols 96, 98, 100
 - HTTP 92
 - TCP 92
 - UDP 91
 - proxy authentication for 74
 - quality of streams 48
 - QuickTime
 - NetCache support for 45
 - UDP ports and 56
 - RDT 81
 - RealFlash 81
 - RealNetworks, supported RealMedia MIME types 81
 - RealPix 81
 - RealProxy restrictions 55
 - RealText 81
 - RTSP

- IP spoofing restrictions for 88
- RealSystem support 45
- RTSP, RealSystem support 45
- server-side HTTP encapsulation 102
- streaming server connections 77
- supported media players 46
- supported protocols 45
- switches for
 - collecting port statistics 87
 - minimum software versions 45
 - recommended duplex settings 47
- TCP
 - NetCache support for 77
- TCP, using for server connections 77
- transparency ports 53, 54, 56
- troubleshooting
 - CLI commands for 93
- UDP
 - NetCache support for 77
- Windows Media metafile rewriting 65
 - request flow 66
- streaming media cache
 - as an accelerator 35
 - authentication for 74
 - caching methodology 83
 - caching restrictions 81
 - client authentication 74
 - client logging information
 - consistency of options 71
 - forwarding for proxied streams 79
 - forwarding for split streams 79
 - client-side HTTP encapsulation 102
 - configuration
 - strategies 47
 - troubleshooting 94
 - troubleshooting with CLI 93
 - configuration planning 61
 - connections 77
 - ContentDirector and 50, 82
 - dynamic stream switching 79
 - ejecting locked objects 70
 - how objects are stored 83
 - limiting client connections 48
 - log forwarding 79
 - consistency of options 71

- MMS
 - NetCache configuration for 45
 - monitoring information 76
 - object expiration 84
 - object management 84
 - objects and 81
 - overloading 48
 - port settings 42
 - RealNetworks, restrictions 81
 - server-side HTTP encapsulation 102
 - supported vendors 81
 - transparent proxying for 53, 54, 56
 - UDP requests 77
 - Web filtering restrictions 42
- streaming server
 - client authentication 74
 - logging information 71
 - relationship to streaming accelerator 118

T

- TCP, streaming media
 - NetCache support for 77
 - using for server connections 77
- Time To Live (TTL)
 - for Web objects 12, 14
 - writing rules 14
- transparent proxying
 - access to NetCache appliances 47
 - planning summary 39
 - streaming media
 - recommended method 47
- transports
 - streaming media
 - HTTP 92
 - TCP 92
 - UDP 91
- troubleshooting, streaming media 93, 94
- tuning NetCache
 - hit rate, Web cache
 - improving 29, 30, 31
 - NNTP bandwidth savings, how calculated 163
 - performance
 - limiting number of access controls 31
 - performance, Web cache

- adding network interfaces 30
- response time for news requests 162

U

- UDP and streaming media 77
- URL byte hit rate
 - compared with URL object hit rate 30
- URL object hit rate
 - compared with URL byte hit rate 30

W

- Web
 - objects
 - making objects uncacheable 10
 - specifying which objects are cached 8
 - which objects are cached 8
- Web accelerator
 - distributed 131
 - DNS configuration for 118
 - examples 128
 - handling all requests 119
 - limited access from another company 142
 - multiple accelerators 131
 - multiple servers and one accelerator 138
 - splitting requests with Web server 120
 - L4 switch in front of 135
 - L7 switch in front of 135
 - limited access from a partner 142
 - outside the firewall 128
 - Server Load Balancer in front of 135
- Web cache
 - aborted requests 17
 - caching objects with cookies 30
 - Complete Fetch option 17

- configuration of basic options 18
- control headers 13
- cookies 9
- creating TTL rules 14
- default object TTL 13
- default settings 18
- defining noncacheable objects 9
- explicit expiration time 16
- hit rate
 - byte hit rate 30
 - improving 29, 30
 - not applicable to news requests 162
- how long objects remain in 14
- how NetCache sends requests 162
- If-Modified-Since query 12
- making content uncacheable 10
- performance
 - adding network interfaces 30
 - disabling Log FQDN option to improve 30
 - limiting number of access controls 31
- planning for configuration
- proxy port settings 19
- specifying which FTP objects are cached 23

- Web objects
 - making objects uncacheable 19
 - refresh rate 16
 - specifying objects not cached 24
 - Time To Live (TTL) 12, 14
 - TTL patterns for 14
- Web server
 - relationship to Web accelerator 118
- Windows Media metafile rewriting 65
 - example of 65
 - request flow 66