

NetCache™ 5.4

Client Monitoring and Control for Streaming

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 210-00356
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	vii
Chapter 1	Overview of NetCache e-Commerce Technologies	1
	Available NetCache e-Commerce technologies	2
	Determining your requirements	5
	About protocols in the NetCache e-Commerce technology suite	6
	About real-time monitoring and control with RTMON	7
	About authentication components	10
Chapter 2	Planning for RTMON	13
	Planning for RTMON application server group configuration.	14
	Planning for data that RTMON can send to application servers	17
	Planning for your application servers that use RTMON	19
Chapter 3	RTMON API	21
	User event message syntax	22
	RTMON user event types.	23
	RTMON user attributes.	24
	Status events	28
Chapter 4	Planning for the use of RADIUS extensions	29
	About IP authentication.	30
	Scenario: IP authentication with user name validation.	33
	Scenario: Using the IP authentication feature for access control by URL . . .	36
Chapter 5	API—NetApp RADIUS extensions	41
	Overview of RADIUS transactions.	42
	Standard RADIUS attributes that NetCache sends	45

NetApp vendor-specific attributes 47

Index 53

Preface

Purpose of this guide

This guide describes the NetCache™ features for e-Commerce that are available for use with application servers. It provides information for the following supported NetCache platforms: C1100, C1105, C1200, C2100, C3100, and C6100.

Who this guide is for

This guide is for

- ◆ Architects and planners who need to determine
 - ❖ The most appropriate NetCache strategy to use for e-Commerce
 - ❖ The configuration required on the NetCache appliance to support e-Commerce
- ◆ Developers who need to program their application servers

Obtaining additional deployment assistance

Network Appliance™ sales engineers provide assistance by helping you to evaluate your requirements, determine how many and what type of NetCache appliances you need, and determine how to deploy your NetCache appliances.

You can find additional information related to NetCache on the NOW Web site (<http://now.netapp.com/>).

About this chapter

This chapter describes the NetCache e-Commerce technologies suite and provides information to help you determine the technologies that best meet your requirements.

Topics in this chapter

This chapter contains information about the following topics:

- ◆ “[Available NetCache e-Commerce technologies](#)” on page 2
- ◆ “[Determining your requirements](#)” on page 5
- ◆ “[About protocols in the NetCache e-Commerce technology suite](#)” on page 6
- ◆ “[About real-time monitoring and control with RTMON](#)” on page 7
- ◆ “[About authentication components](#)” on page 10

Available NetCache e-Commerce technologies

Overview of NetCache e-Commerce technologies

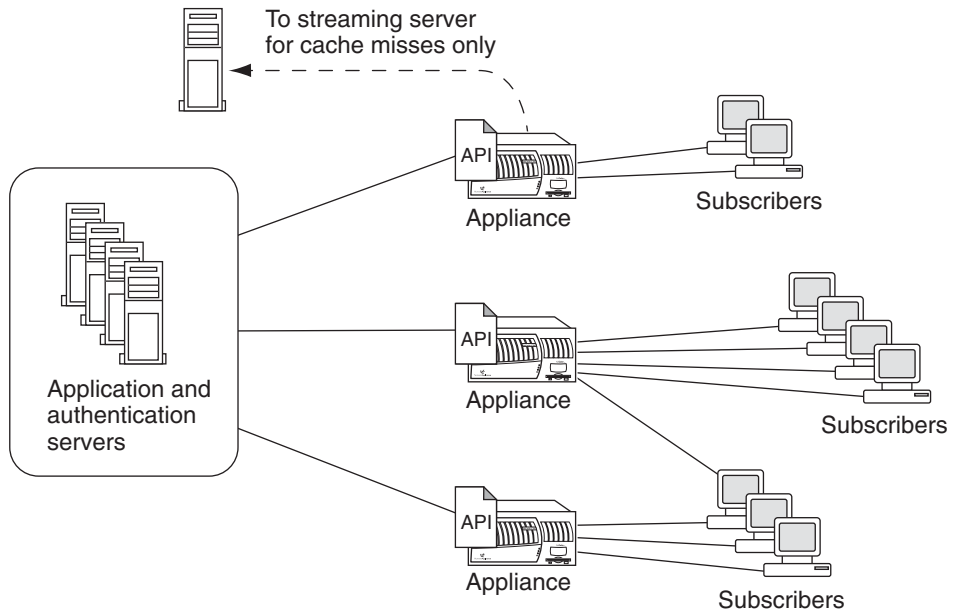
NetCache e-Commerce technologies provide features that enable you to monitor, manage, and control user streaming activity. These technologies are fundamental to providing new opportunities for generating revenue.

The following table shows some examples of how NetCache e-Commerce technologies can benefit service providers and enterprises.

If you are...	Example of use
A service provider	◆ User activity data for billing applications ◆ Pay-Per-View video on-demand and live streaming
An enterprise	◆ Accounting by department

NetCache e-Commerce technologies are based on a set of protocols on the appliance that enable the appliance to work in conjunction with application servers to monitor and control user streaming activity. (An *application server* is defined in this document as a server such as a monitoring server or a billing server.) Application server developers program their servers to use the NetCache APIs, which enable the servers to communicate with the appliance. Configuration on the appliance enables the appliance to communicate with application servers.

As the following illustration shows, the appliance works in a partnership with application servers in monitoring, controlling, and accounting for user streaming activity.



Benefits of using NetCache solutions for application servers

Caching is an already-proven technique for saving bandwidth and delivering high-quality streams to end users. NetCache provides a variety of benefits in addition to caching objects to support e-Commerce deployments, including the following:

- ◆ NetCache maps user events from different media players into a common format that represents a “universal media player.”
As a result, application servers can be programmed for a single generic media player that includes generic characteristics for type and version. No reprogramming of an application server is required when new media players are introduced to the market or a type of media player is changed.
- ◆ Application developers write to a single NetCache API for current and future streaming protocols.

Previously, application server developers have had to write to several separate streaming server APIs (for example, to one interface for MMS, to one interface for RealNetworks® RTSP, and to one interface for QuickTime™ RTSP). This process has been complex and time-consuming. With the NetCache approach, application server developers write to only *one* NetCache API for all streaming protocols. When streaming protocols are

added later, no reprogramming of the application servers is needed to accommodate the protocols.

- ◆ Extensive authentication support (See “[About protocols in the NetCache e-Commerce technology suite](#)” on page 6.)
- ◆ The ability to monitor user streaming activity in real time using XML-based technology
- ◆ The ability for the appliance to communicate with multiple application servers, thereby supporting redundancy
- ◆ Seamless integration with existing accounting applications and Digital Rights Management (DRM) technologies
- ◆ Scalable deployment that accommodates your organization’s needs as your user base grows
- ◆ Minimal impact to your network

Determining your requirements

Where to start

The protocols that NetCache provides in its e-Commerce technology suite provide a great deal of flexibility. You need to

- 1. Decide your monitoring, management, and control requirements
- 2. Determine which protocols that NetCache provides meet your requirements

Monitoring and control components

Organizations that deploy application servers are commonly interested in including one or more of the following components in their deployment schemes.

Component	Defined as...
Authentication	Who the user is
Authorization	What the user can do
Accounting	What the user did

Not all the protocols in the NetCache e-Commerce technology suite include all the components listed in the previous table, nor do all protocols provide the same level of functionality of a particular component. You can use one or more protocols to achieve the results that you need.

For example, is it sufficient for the appliance to push real-time data to your application server but provide no other controls (that is, provide only an accounting function)? Do you also need authentication?

The remainder of this chapter helps you understand the protocols in the e-Commerce technology suite that are available and the monitoring and control components that are associated with each protocol.

About protocols in the NetCache e-Commerce technology suite

Available protocols

NetCache provides the following protocols in its e-Commerce technology suite:

- ◆ Real-Time Monitoring protocol (RTMON)
- ◆ NetApp RADIUS extensions (NetCache vendor-specific RADIUS extensions)

Summary: monitoring and control components, by protocol

The following table summarizes the monitoring and control functions that are associated with each protocol. Each protocol and the level of monitoring and control provided with the protocol are discussed in later sections of the chapter. You can mix and match components; for example, if you are using RTMON, you can add a component that provides authentication.

Protocol	Implements authentication?	Implements accounting?	Implements authorization?
Real-time monitoring (RTMON)	No	Yes	No
RADIUS with extensions	Yes	No	Yes (partial)
RADIUS without extensions	Yes	No	No
NTLM, LDAP, NetCache local database	Yes	No	Yes (through groups and ACLs)

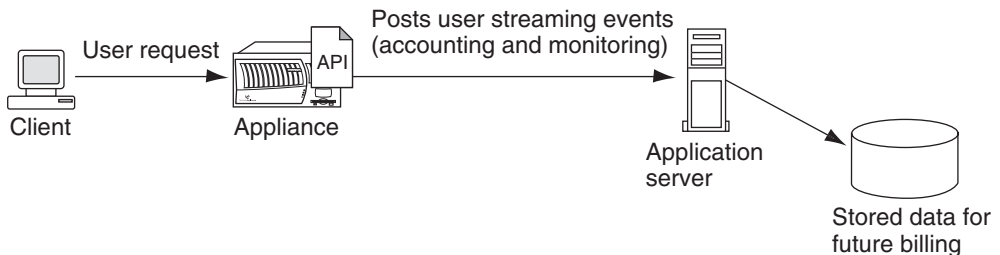
About real-time monitoring and control with RTMON

Overview of RTMON The Real-Time Monitoring protocol (RTMON) is a NetCache proprietary XML-based protocol that is used for monitoring user streaming activity in real time. 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.

Real-time monitoring is useful when you need your application servers to receive data while the activity occurs, instead of a reporting program extracting data from stored log files at a later time.

Real-Time Monitoring protocol (RTMON)

RTMON sends user streaming event data to specified application servers in real time. Data is sent using HTTP POST, at intervals that you specify. RTMON closes the connection at the end of each posting interval.



As the previous illustration shows, RTMON on the appliance listens for user streaming events from clients and sends that data to application servers. RTMON provides only monitoring. RTMON does not support application servers working with the appliance to exert any control over user streaming activity.

How an application server uses the monitored data depends on the type of server. For example:

- ◆ An application server might store the data in a database for use for future billing, as shown in the previous illustration.
- ◆ An application server in an ISP that wants to control the amount of time that a user is dialed in and streaming could send data that NetCache sends it to a database server that contains information about the amount of time users have been viewing. The database server could communicate with terminal

servers to which users are logged in to close connections of users who have been dialed in and streaming too long. In this case, the application server does not interact with the appliance to control user activity. Instead, the server uses the data from the appliance to interact with other devices that provide controls.

Feature summary: RTMON

The following table summarizes various aspects of how RTMON works.

Feature aspect	RTMON
Data sent between the appliance and application server	◆ Status information ◆ Real-time monitoring of the following user streaming events: ❖ Connect ❖ Play ❖ Pause ❖ Set-bitrate ❖ Disconnect ❖ Dropped events Detailed data is available about each of these events. See Chapter 3, “RTMON API,” on page 21 for details about events the appliance provides.
Application server control over streams	None. Only monitoring functions are available.
How the protocol communicates with the application server	◆ Push RTMON sends messages about specified user activity to application servers by using HTTP POST. Connections are reestablished at the start of each (configurable) posting interval. TCP socket is also supported. See “What occurs during failover” on page 15 for more information.

Feature aspect	RTMON
Implication of the communication model on technology that can be used by developers	Application server developers can implement Web server CGI scripts or Apache plug-in applications.
Simple Access Object Protocol (SOAP)	Not supported.
Configuration on the NetCache appliance	Configuration on the appliance determines which events and data are posted.
Schema (how the XML data is presented)	Data is sent through XML attributes. Example: <User Event-"play"CID=1 .../>

About authentication components

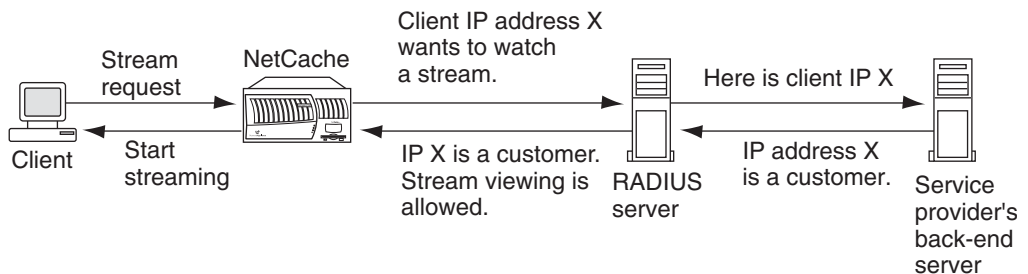
Enhanced authentication with RADIUS with extensions

NetApp RADIUS extensions enhance the capabilities of standard RADIUS authentication. These NetApp vendor-specific extensions provide NetCache the ability to authorize requests based on IP address and URL, and to rewrite URLs and server names.

The primary goals when using NetApp RADIUS extensions in an e-Commerce deployment are for NetCache to obtain the user's identity but not to prompt for it and to simultaneously inform the NetCache appliance about authorization rules. The assumption in this case is that the IP address is known.

For example, many service providers maintain a mapping of IP addresses to users. A NetCache appliance queries a RADIUS server about the access permissions granted to a client that has a given IP address (referred to as *RADIUS IP authentication*).

The following illustration shows successful authentication based on client IP address; that is, the server did not have to prompt the client software for a user name and password.



As the previous illustration shows, the appliance sends the client IP address and requested URL to the RADIUS server. The RADIUS server passes the client IP address to the service provider's back-end server, which stores a list of customers. Because the IP address can be matched to the client IP address, the client is approved to view the stream.

The RADIUS server can allow the request (as shown in the previous illustration), deny it outright, or direct the NetCache appliance to prompt the user for authentication if the RADIUS server cannot match an IP address with a user.

Caution

NAT and firewalls can cause problems with RADIUS IP authentication.

Requirements for using NetApp RADIUS extensions are as follows:

- ◆ The scheme can be used only if a RADIUS server is deployed.
- ◆ The RADIUS server must be customized to use NetApp vendor-specific attributes for NetCache.
- ◆ Several options must be configured on the NetCache appliance. (See Chapter 2, “[Planning for RTMON](#),” on page 13 for more information.)

RADIUS extensions are supported only for streaming media.

The following table summarizes the authentication, accounting, and authorization capabilities that are available with RADIUS extensions.

Component	Description
Authentication	Authentication is IP-based.
Accounting	The packet that the appliance sends to the RADIUS server contains the URL that the user requested. The RADIUS server extracts data about the URLs that your users have requested. That data can then be sent to another application server that can prepare the data for your analysis. The accounting capabilities provided by RADIUS extensions are more limited in scope than the capabilities provided by RTMON and provides “static” information. No real-time communication session is set up between the appliance and the RADIUS server for monitoring user activity, as is the case with RTMON. Using RADIUS extensions and RTMON together provides a robust solution for e-Commerce.
Authorization	Authorization functions are provided by the application server.

Standard authorization methods

If the method that you are choosing to monitor and control your e-Commerce activities does not provide authentication (for example, RTMON), you can add an authentication component by selecting one of the standard authentication features that NetCache supports, or by using RADIUS extensions (if your network has a RADIUS server). NetCache supports the following standard authentication features:

- ◆ Mechanisms for validating a clear-text credential
 - ❖ RADIUS authentication server

If your network has a RADIUS server, standard RADIUS authentication will provide basic authentication in which the user is prompted for a user name and password. If, however, you are interested in promptless user name and password authentication, you must use NetApp RADIUS extensions.
 - ❖ LDAP authentication server
 - ❖ NetCache local database
- ◆ Password encryption schemes
 - ❖ NTLM (NT LAN Manager)
 - ❖ Kerberos (Windows 2000)

NTLM and Kerberos can be used only with clients that support them.

The *Security Guide* and the Appliance Manager online Help discuss the standard authentication methods that NetCache supports.

About this chapter

Setting up RTMON requires both configuring a NetCache appliance and programming one or more application servers to use the RTMON API on the appliance. This chapter describes the planning considerations for deploying RTMON on the appliance and on application servers.

See Chapter 3, “[RTMON API](#),” on page 21 for information you need to program the API.

Topics in this chapter

This chapter contains information about the following topics:

- ◆ “[Planning for RTMON application server group configuration](#)” on page 14
- ◆ “[Planning for data that RTMON can send to application servers](#)” on page 17
- ◆ “[Planning for your application servers that use RTMON](#)” on page 19

Planning for RTMON application server group configuration

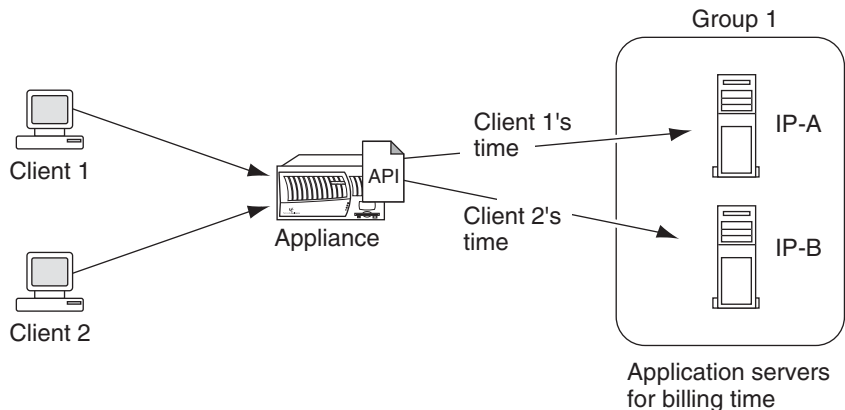
What you need to plan for

When you configure RTMON to work with application servers, you must

- ◆ Identify the application servers to which the appliance will send data
- ◆ Identify the data you want the appliance to send to specific groups of application servers

Application server group defined

When you configure RTMON to work with application servers, you need to configure one or more logical *application server groups* in the appliance RTMON configuration.



Application server groups have the following primary purposes:

- ◆ To enable you to identify which data you want the appliance to send to which application servers

Each group is identified by a logical group name of your choosing. You could, for example, assign a group of billing servers a name that would identify them as billing servers. Mappings of the group name to IP addresses is handled by DNS.

You might have different types of application servers that need the same data. Alternatively, you might have two different types of application servers that require different data. You define, by logical group, the data that the appliance is to send to servers in the group. See [“Planning for data that RTMON can send to application servers”](#) on page 17.

- ◆ To provide a failover scheme in case of application server unavailability
- ◆ To provide a load-balancing scheme

Data that each server in an application server group receives

When multiple application servers are members of the same logical application server group, each server receives the same user events and user attributes. However, NetCache always sends a specific user's data to the same application server, which balances the load over the server as described for load balancing. See also "[What occurs during failover](#)" on page 15.

How load balancing works

To balance the load of data across application servers in the same application server group, NetCache uses a weighting technique based on a history of connection round trips. Server CPU load and other server resources are not considered, nor are differences such as bandwidth used by the streams that specific clients are watching.

Initially, all servers have the same chance of being selected. If a server is consistently slower in responding to the appliance over time, the chance decreases that the appliance will send data to the application server.

In practice, this scheme has resulted in a fairly even distribution of data across application servers in an application server group. However, you should monitor the load on your application servers resulting from data pushed by the appliance and make adjustments to your server size or number of servers as necessary.

What occurs during failover

When you define an application group, you are instructing the appliance to distribute the data over all servers in the group. If one server becomes unavailable, NetCache continues to automatically distribute the data over the remaining available servers in the group.

The following occurs if an application server in an application server group becomes unavailable:

- ◆ RTMON sends data about new users to available servers in the same application server group.
- ◆ For users whose data was being sent to the server that became unavailable, the following occurs:
 - ❖ If HTTP POST and buffering are configured on the appliance, the appliance attempts to resend the events in the buffer after the server becomes available again.

The NetCache appliance retains the data until the logs are deleted. By default, NetCache buffers data until it runs out of disk space. A rough estimate of when the appliance would run out of disk space is when the remaining disk space falls below 10 percent of the total NetCache volume size.

- ❖ If TCP socket is configured on the appliance, no buffering will occur. Therefore, no data is available for the appliance to resend to an application server after the server becomes available again.

Network Appliance recommends that you set up HTTP with buffering so that the appliance can buffer data for resending when the server becomes available.

Number of application servers communicating with NetCache

You can add application servers to application server groups as necessary to increase reliability and scalability. Network Appliance recommends that you do not exceed 10 application servers communicating with a NetCache appliance.

Considerations for defining application server groups

Consider the following when defining your application server groups:

- ◆ You might want to include servers on different networks in the same application server group. That way, if one network becomes unavailable, the other servers in the application server group can handle the monitoring services.
- ◆ You might want to use dedicated servers to reduce the possibility of the servers becoming bottlenecks.
- ◆ You can group servers by the type of data they need the appliance to send to them. See [“Planning for data that RTMON can send to application servers”](#) on page 17.

Planning for data that RTMON can send to application servers

Requirements for identifying data to be sent

For each logical application server group, you must identify in the RTMON configuration on the appliance the data that the appliance is to send to the application servers in that group. When you configure RTMON on an appliance, you identify for each application server group the user events and user attributes that the appliance is to send to the servers in the group. The application servers in the group receive data only about those user events and user attributes from the appliance. All servers in the application server group receive all the user events and user attributes that you specified for the group.

Your application server developers must program the application servers in the application server group that you have defined on the appliance to monitor for the user events and user attributes of interest that the appliance sends. See Chapter 3, “[RTMON API](#),” on page 21 for information about server setup.

User events and user attributes defined

User events provide information about the streaming activity that the user is performing, for example, viewing (playing) a video stream. User attributes provide details about each user event, for example, which user (client) is viewing the stream and how long the user has been viewing the stream.

Available user events

The appliance can send any or all of the following user events to servers in an application server group:

- ◆ Connect
- ◆ Play
- ◆ Pause
- ◆ Disconnect
- ◆ Dropped events
- ◆ Set-bitrate

See “[RTMON user event types](#)” on page 23 for a description of each user event.

Available user attributes

The appliance can send to servers in an application server group any or all of the user attributes described in “[RTMON user attributes](#)” on page 24:

The value shown, for example, CONNECTED-TIME, is the XML value that your application server developer must use when programming the application server to use the RTMON API. You can use that value to look up a detailed description for each user attribute in Chapter 3, “[RTMON API](#),” on page 21.

Status events

You can configure the appliance to send status data to servers in an application server group. If the appliance sends status data, it sends

- ◆ The appliance IP address
- ◆ The current number of connected clients for each protocol
- ◆ The average CPU usage over the interval set over the post
- ◆ Average streaming downstream bandwidth, in kilobytes per second

Heartbeat events

You can configure the appliance to send heartbeat events at specified intervals. The appliance then sends a heartbeat event per period when no other events are pending. Configure your appliance to send heartbeat events to your application servers if you are interested in liveness reports indicating from the appliance.

A heartbeat event contains

- ◆ A timestamp
- ◆ The appliance IP address

Planning for your application servers that use RTMON

Adding an additional application server group for redundancy

Your NetCache deployment will already have been set up to ensure maximum reliability between clients and the appliance. However, to ensure optimal functioning of your e-Commerce deployment, you need to plan for how to handle application server failure. RTMON's appliance configuration of logical application server groups provides a failover scheme. However, you must have enough application servers to support the load if one or more application servers become unavailable. Network Appliance recommends that you add at least one more server group than you would typically need to handle essential monitoring functions, for example, for billing services.

Programming of application servers

Your developers must program the application servers to use the RTMON protocol on the appliance. The RTMON XML attributes available for the developers are documented in Chapter 3, "[RTMON API](#)," on page 21.

About this chapter

This chapter provides information about the XML attributes used in the RTMON API on the NetCache appliance. Use information in this chapter to program your application servers to use the RTMON protocol on the appliance for real-time monitoring of user streaming events.

Topics in this chapter

This chapter contains information about the following topics:

- ◆ [“User event message syntax”](#) on page 22
- ◆ [“RTMON user event types”](#) on page 23
- ◆ [“RTMON user attributes”](#) on page 24
- ◆ [“Status events”](#) on page 28

User event message syntax

Syntax

The syntax of an RTMON user event message is as follows:

```
<User EVENT="{event-type}" {user-attributes} [{event-attributes}]/>
```

Example of an RTMON message

An example of an RTMON message follows. The user events and user attributes shown in the message are described later in this chapter.

```
<response>

<User EVENT="connect" NIP="10.5.8.26" SIP="172.3.20.5"
CIP="10.5.50.75" CID="3" STREAM="g2video.rm"
MTYPE="R" DT="1001754931" GUID="00000000-0000-0000-0000-
000000000000" URL="rtsp://real-server.some.com:554/g2video.rm"
QS="&bytes-transferred=819&buffer-time=1&packets-lost=0"/>

<User EVENT="disconnect" NIP="10.5.8.26" CID="3"
STREAM="g2video.rm" MTYPE="R" DT="1001754934"
GUID="00000000-0000-0000-0000-000000000000" URL="rtsp://real-
server.some.com:554/g2video.rm"

<User EVENT="dropped-events" NIP="10.56.17.185" CID="62"
STREAM="file.wmv" MTYPE="W"
DT="1003298424" GUID="503379d0-2346-11d3-b63f-00a0c9eeea28"
URL="mms://server.mycorp.com:1755/movies/file.wmv"
QS="&bytes=1389256&buftime=113&pktloss=0"/>

</response>

<response>

<Heartbeat NIP="10.5.8.26"/>

</response>
```

RTMON user event types

Programing your application servers

You will need to program your application servers to process the user event types and user attributes that your server applications need. The user attributes (described in “[RTMON user attributes](#)” on page 24) provide details about all of the event types that your server might use.

Event types that can be sent to servers

The following table describes all the RTMON event types that the appliance can send to application servers. Whether a specific application server receives all user event types depends on how the NetCache administrator has configured RTMON on the appliance. The NetCache administrator specifies, by logical application server group, the RTMON event types that will be sent to servers in that group.

User event type	Description
connect	The client has established the connection and started playing the stream.
disconnect	The client stopped playing the stream and closed the connection.
dropped-events	When the maximum buffer limit is reached, currently connected clients log a “buffer full” message and drop all subsequent events except the Disconnect event.
pause	The client paused the stream while it was being played.
play	The client is in the process of playing the stream.
set-bitrate	The client requests a change in bit rate.

RTMON user attributes

Programing your application servers

You will need to program your application servers to process the user event types and user attributes that your applications need.

User attributes that can be sent to servers

The following table describes all the RTMON user attributes that the appliance sends to application servers. User attributes provide details about the user events that occur. Whether a specific application server receives all user attributes depends on how the NetCache administrator has configured RTMON on the appliance. The NetCache administrator specifies by logical application server group the RTMON user attributes that will be sent to servers in that group.

All RTMON user attributes are delivered in each user event.

User attribute	Description
CID	Client ID. For each unique client stream, NetCache generates a CID.
CIP	Client IP address
CONNECTED-TIME	Amount of time the client has been connected.
DELIVERY-BITRATE	Bit rate at which the stream is being delivered, in kilobytes per second.
DROPPED-PACKETS	Total number of client-side packets dropped.
DT	Time stamp of the event since the UNIX Epoch time (00:00:00 GMT, January 1, 1970), for example, “1003298424”. DT is suitable for input to C or PERL “gmtime” and “localtime” routines.
DT-USEC	Microsecond portion of the event timestamp, that is, microseconds past DT.
ENCODING-BITRATE	Bit rate at which the stream was encoded, in kilobytes per second.

User attribute	Description
GUID	Globally Unique ID string that a media player might send to the appliance to identify itself. GUIDs follow a specific format, for example, “503379d0-2346-11d3-b63f-00a0c9eeca28.” Not all streaming protocols include GUIDs, and GUIDs for different protocols might vary in length. The GUID should be treated as an opaque string that can be used to track a particular client by protocol. Currently, of the protocols that the appliance supports, QuickTime does not include GUIDs. GUIDs from RealPlayer that consist of all zeroes indicate that the user has disabled the sending of GUID information in the player preferences.
HIT-BYTES	Number of bytes sent to the client that were served from the cache (a cache hit).
LIMIT-BANDWIDTH	Bit rate that the client requested but was refused because Admission Control is in effect on the appliance for that client and dedicated bandwidth was unavailable. Admission Controls ensure that streaming requests at specific bandwidths are accepted only when it is possible to deliver a stream with predictable quality.
MISS-BYTES	Number of bytes that the appliance sent to the client from the origin server (a cache miss).
MTYPE	The player type: Q = QuickTime R = RealPlayer W = Windows Media Player
NIP	Appliance IP address.
QS	The query string portion of the URL (everything after the ‘?’ in the URL).

User attribute	Description
PLAY-TIME	The amount of time the client has been playing the stream.
PROXY-USERNAME	User name supplied by the client when authenticating to the appliance.
SESSION-ID	A session ID is sent by the RADIUS server if the RADIUS IP authentication feature is in use. The application server can use the SESSION-ID to share information (or state) with the RADIUS server.
SIP	Origin server IP address.
STREAM	The name of the media file in the URL, for example, "file.wmv".
TOTAL-BYTES	Total number of bytes delivered to the client.

User attribute	Description
TRANSPORTS	Shows the following information about connections that exist between the appliance and the client and the appliance and the server: ◆ End points of the connection <code><type>client server cache</type></code> ◆ IP addresses and ports <code><ip>i.b.c.d</ip> <port>1234</port></code> ◆ The protocol used for the connection, either TCP or UDP <code><protocol>tcp udp</protocol></code> ◆ The type of connection, which can be ❖ control (“control”) ❖ audio stream (“audio”) ❖ video stream (“video”) ❖ retransmits and acknowledgements (“feedback”) The syntax is: <pre> <media> <type>audio</type> <type>feedback</type> ... </media> </pre>
URL	The URL that the client sent, for example, “mms://server.corpA.com:1755/movies/file.wmv”.

Status events

Available NetCache status events

NetCache administrators specify, by logical application server group, whether the appliance will send status information to the server in the group. If status data will be sent, the appliance sends all data in the following table.

Status attribute	Description
NIP	NetCache IP address
NUM-RTSP-QT	Current number of connected QuickTime clients
NUM-RTSP-REAL	Current number of connected RealSystem clients
NUM-MMS	Current number of connected MMS clients
NUM-RTSP-UNKNOWN	Current number of connected unknown RTSP clients
AVG-CPU-LOAD	Average percentage of CPU usage over the interval set on the NetCache appliance.
AVG-DOWNSTREAM-BW	Average streaming downstream bandwidth (to clients in kilobytes per second)

Status event message example

The following example shows a Status event message that a NetCache appliance might send to an application server:

```
<Stats NIP="10.56.17.185" NUM-RTSP-REAL="1" NUM-MMS="3" AVG-CPU-LOAD="4%" AVG-DOWNSTREAM-BW="29" />
```

Heartbeat message example

The following example shows a Heartbeat message that a NetCache appliance might send to an application server:

```
<Heartbeat NIP="10.56.17.185" DT="12345" />
```

About this chapter	This chapter provides information to help you understand how you can use RADIUS extensions in your e-Commerce deployments.
Topics in this chapter	This chapter contains information about the following topics: ◆ “About IP authentication” on page 30 ◆ “Scenario: IP authentication with user name validation” on page 33 ◆ “Scenario: Using the IP authentication feature for access control by URL” on page 36
Procedures and option descriptions	See the Appliance Manager online help for configuration procedures and option descriptions.
Location of configuration options	Options for configuring IP authentication are in the following location in the Appliance Manager: Setup tab > Authentication > RADIUS > Extensions tab
Planning for programming the RADIUS server	See Chapter 5, “ API—NetApp RADIUS extensions ,” on page 41 for information about each NetApp vendor-specific attribute and how these attributes are passed in a RADIUS transaction.

About IP authentication

Comparing IP authentication and standard RADIUS

The following table provides information about the IP authentication protocol and standard RADIUS that you need to consider when determining whether you can use IP authentication in your organization.

Consideration	IP authentication (through RADIUS extensions)	Standard RADIUS
Protocols supported	Streaming media only	All
How requests are sent	NetCache checks <i>all</i> requests with identified RADIUS servers.	NetCache sends a request to identified RADIUS servers only if the client sends an authorization header.

The IP authentication protocol provides the interface between a NetCache appliance and an application server. A NetCache appliance is aware of RADIUS servers but it is not aware of application servers.

Implication of all requests being sent to RADIUS servers

If you enable IP authentication, the NetCache appliance queries the RADIUS servers you have identified for every request it receives. You must ensure that the RADIUS servers that you have identified in the appliance configuration can handle the load of requests that the appliance will send to them.

Some organizations dedicate specific NetCache appliances and RADIUS servers for their e-Commerce deployments. Doing so provides a way to limit the volume of traffic sent to the appliances that would be forwarded to the RADIUS servers.

Why IP authentication is only for streaming protocols

For streaming requests, the appliance behavior of sending every request to the RADIUS servers should not be a problem for a typical RADIUS server. The reason is that with streaming, an appliance receives a relatively low number of requests per second. A NetCache appliance sends a streaming request to the RADIUS server only once per stream, regardless of the duration of the stream.

In contrast, an appliance receives a large number of HTTP requests per second. If an appliance on which IP authentication was enabled was handling HTTP requests, the appliance would query each identified RADIUS server to validate each HTTP request. Consequently, the RADIUS server probably would become overloaded by the large volume of HTTP requests.

Summary of prompting activity with IP authentication

When IP authentication is enabled, the NetCache appliance first attempts to authenticate based on client IP address. The following table summarizes the NetCache behavior after the appliance receives specific responses from the RADIUS server.

If...	Then...
Authentication by client IP address is successful.	The client is not prompted for a user name and password, and NetCache uses credentials supplied by the RADIUS server.
Authentication by client IP address is <i>not</i> successful.	The RADIUS server instructs NetCache to request a user name and password from the client software. If the user name and password are not valid, the client is denied.
Authentication by user, password, or the client IP address is successful but the user is denied because the user is not authorized (for example, because the account has insufficient funds or it might have expired). Authorization is determined by the application (see the scenarios later in this chapter for more details).	The client is denied.

Options for configuring IP authentication

On the appliance, you must configure both basic RADIUS options and specific options. All RADIUS options are on the Authentication - RADIUS page.

Basic options on the General tab: Configure the following:

- ◆ RADIUS Enable
- ◆ RADIUS Servers

Options on the Extensions tab specific to IP authentication: The following options are available for IP authentication.

If you want to...	Use these options on the appliance...
Enable the IP authentication feature	These options are always required: ◆ Use IP Authentication Protocol ◆ IP Authentication Password
Ensure that the media player does not use a user and password that it has stored	This option is optional: ◆ Always Prompt for User Name and Password
Send the URL with the request	This option is optional: ◆ Send URL See “ Scenario: Using the IP authentication feature for access control by URL ” on page 36 for an example of using this option with IP authentication. This option can be used with standard RADIUS also.

Scenario: IP authentication with user name validation

About this scenario

The goals for this scenario are as follows:

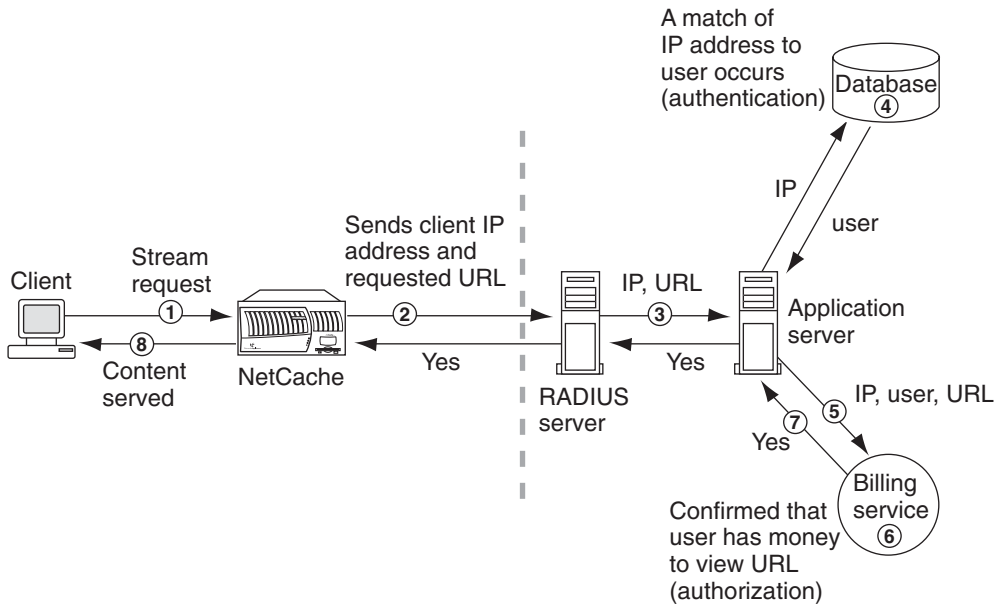
- ◆ Authenticate the user without prompting
- ◆ Authorize the user by client IP address, the user associated with that IP address, and the requested URL

If a problem prevents the RADIUS or application server from obtaining a mapping of user to IP address, the RADIUS server can inform the NetCache appliance to request a user name and password from the client software.

In this scenario, the NetCache appliance is aware of the identity of the user who is making a request. For a deployment in which the NetCache appliance is unaware of users' identities, see "[Scenario: Using the IP authentication feature for access control by URL](#)" on page 36.

Example 1: Successful authentication and authentication without prompting

The following illustration shows authentication successfully accomplished based on the client IP address—that is, the user is not prompted for a user name and password. In this example, the user is authorized to view the requested stream.



The following table provides details about the preceding illustration.

Stage	Description
1	The user requests a stream.
2	The appliance sends the requested URL and the client IP address to the RADIUS server.
3	The RADIUS server sends the client IP address and requested URL to the application server.
4	The application server attempts to match the IP address that was passed to it to a user name in its database. If no match occurs, the application server informs the RADIUS server, which in turn informs the NetCache appliance that it must inform the client software to prompt for a user name and password.

Stage	Description
5	If a match occurs, the application server sends the IP address, user name, and URL to its authorization component (represented by the billing service in this illustration).
6	The authorization component determines whether the user can be approved to watch the stream.
7	If the authorization component approves the user to watch the stream, the application server and RADIUS server inform the NetCache appliance. The appliance then serves the requested content. If the authorization component does not approve the user's request to watch the content, then the client is denied.

Planning for configuration

Configuration of the NetCache appliance and the RADIUS server involves the following.

Configuration requirements for the appliance: Configure the following on the Setup tab > Authentication > RADIUS page:

- ◆ Basic RADIUS configuration on the appliance
- ◆ Use the IP Authentication Protocol option and IP Authentication Password option (on the Extensions tab of the RADIUS page)

Configuration requirements for the RADIUS server: The RADIUS server developer must program the RADIUS server to

- ◆ Use the password configured in the IP Authentication Password option on the appliance to connect to the appliance
- ◆ Recognize the Auth-type value in Vendor-Netapp-2 and recognize that type as a 1 or 2
- ◆ Send Vendor-Netapp-3 in Access:Reject packets

See Chapter 5, “[API—NetApp RADIUS extensions](#),” on page 41 for more details.

Scenario: Using the IP authentication feature for access control by URL

About this scenario

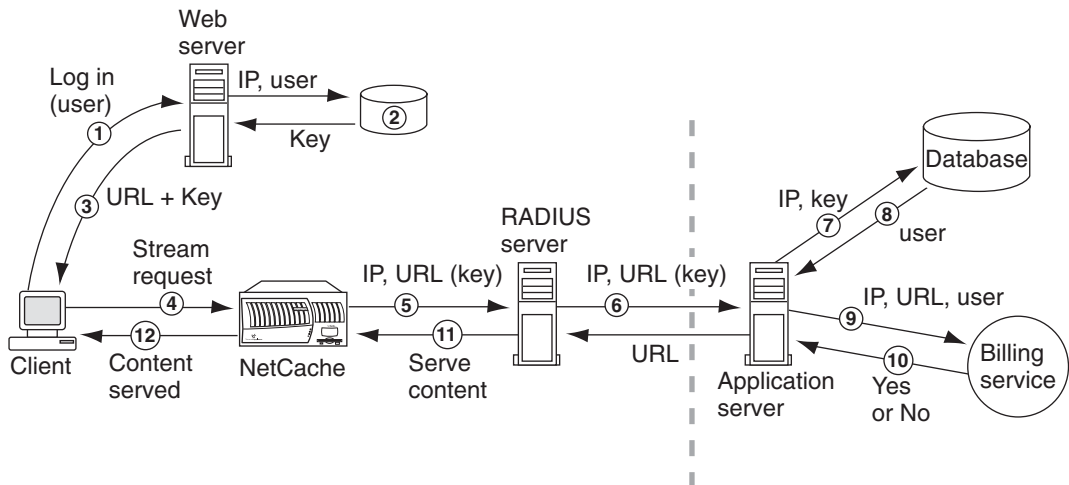
The deployment in this scenario can be used in an organization in which users access URLs that include keys that the application servers map to specific users, for example, to a URL in the form

protocol://host/object?key=...

In this type of scenario, the appliance is configured to send the URL that the user requested to the RADIUS servers. The RADIUS servers and application servers are responsible for all activity related to authentication and determining the URLs that the users can view. All information that the application server needs is encoded in the URL. The NetCache appliance is never aware of the identity of the user who is making a request (unlike the previous scenario), and the user is never prompted for a user name and password. The RADIUS server only informs the appliance about the URL to serve.

Deployment illustrated

The following illustration shows an example in which the application server contains the components to authenticate and authorize users.



The following table provides details about the preceding illustration.

Stage	Description
1	The client logs in to a Web server (or an application server).
2	The Web server uses the IP address and user name to obtain a key from the database.
3	The Web server returns the URL and the key to the client.
4	The client opens the URL, which includes the key.
5	The NetCache appliance sends the client IP address and the requested URL to the RADIUS server.
6	The RADIUS server might process the IP address and URL itself or, as the previous illustration shows, send it on to the application server.
7	The application server obtains the key from the URL and the IP address from the request and checks the database to determine if the key is valid for the IP address. This is the authentication step. Note The key in the URL is meaningful only to the RADIUS server or application server that is deployed in front of the application server. The key is opaque to the NetCache appliance.
8	If the key is valid, the database returns the user name associated with the IP address. If the key is not valid, the user's request is rejected and the user is redirected to a URL other than the requested URL, for example, a stream that you have provided that explains the rejection.
9	If the key is valid, the IP address, user name, and URL are sent to the authorization service, for example, a billing service.

Stage	Description
10	The service attempts to authorize the user. For example, a billing service might determine whether the user's account has enough money to allow the user to view the requested stream. The URL that the application server and then the RADIUS server return to the NetCache appliance is determined by whether the user is authorized. ◆ If the user is authorized, the URL passed to NetCache is for the requested content. ◆ If the user is not authorized, the URL passed to NetCache is of another URL you have provided, for example, the URL of a stream explaining the rejection.
11	The RADIUS server instructs the NetCache appliance to serve the URL to the client.
12	The NetCache appliance serves the content, as instructed by the RADIUS server.

Planning for configuration

Configuration of the NetCache appliance and the RADIUS server involves the following.

Configuration requirements for the appliance: Configure the following on the Setup tab > Authentication > RADIUS page:

- ◆ Basic RADIUS configuration on the appliance
- ◆ Use the IP Authentication Protocol option and IP Authentication Password option (on the Extensions tab of the RADIUS page)
- ◆ Send URL option

The Send URL option enables the appliance to send the URL to the RADIUS server, which enables this type of deployment to function.

Programming requirements for the RADIUS server: The RADIUS server developer must program the RADIUS server to

- ◆ Use the password configured in the IP Authentication Password option on the appliance to connect to the appliance
- ◆ Recognize the Vendor-Netapp-1 attribute
- ◆ Recognize the Auth-type value in Vendor-Netapp-2 and recognize that type as a 1 or 2

- ◆ Recognize either Vendor-Netapp-4 or Vendor-Netapp-5 (not both) to redirect denied users

See Chapter 5, “[API—NetApp RADIUS extensions](#),” on page 41 for more details.

About this chapter This chapter describes the NetApp vendor-specific attributes that are available for use with RADIUS servers.

Topics in this chapter This chapter contains information about the following topics:

- ◆ [“Overview of RADIUS transactions”](#) on page 42
- ◆ [“Standard RADIUS attributes that NetCache sends”](#) on page 45
- ◆ [“NetApp vendor-specific attributes”](#) on page 47

RFC for RADIUS See RFC 2865, “Remote Authentication Dial In User Service (RADIUS).”

Overview of RADIUS transactions

What this section contains

The information in this section helps you understand how to program a RADIUS server for NetApp vendor-specific RADIUS extensions so that your RADIUS servers can

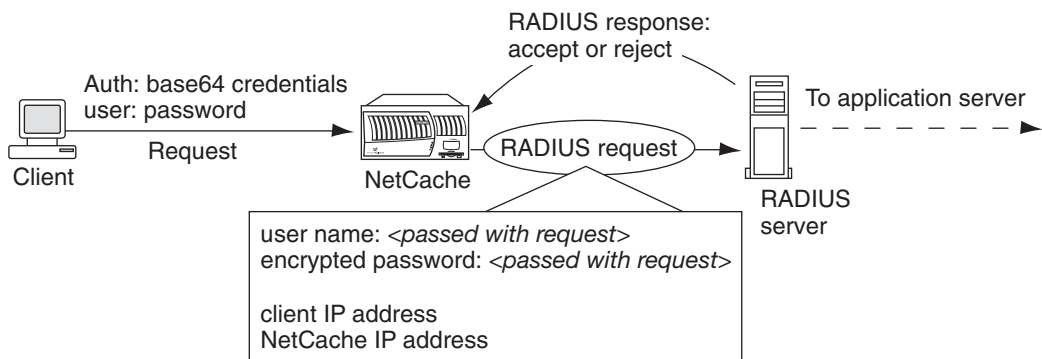
- ◆ Process requests from the NetCache appliance
- ◆ Send the correct response to the NetCache appliance so that the appliance takes the action that you intend

Enabling IP authentication on the appliance informs the appliance to support all NetApp vendor-specific attributes described in this chapter. You determine which attributes to program on your RADIUS server in order to meet your requirements.

To understand what you need to program on your RADIUS servers to support NetApp RADIUS extensions, it is helpful to compare a standard RADIUS transaction and a transaction in which NetApp RADIUS extensions are used.

Standard RADIUS transaction

The following illustration shows a standard RADIUS transaction.

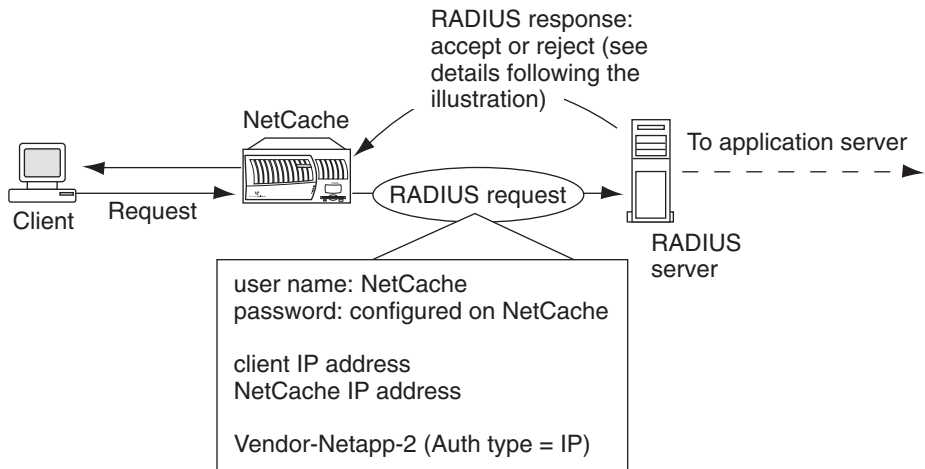


With a standard RADIUS transaction, NetCache

- ◆ Sends a request to a RADIUS server only if the client sends an authorization header
- ◆ Sends a request to a RADIUS server to validate credentials from all client protocols (HTTP, FTP, Gopher, NNTP, RTSP, and MMS)

RADIUS transaction with NetApp vendor-specific attributes

The following illustration shows when NetApp vendor-specific attributes are used during a RADIUS transaction.



When IP authentication is enabled, NetCache sends all requests to the RADIUS server.

As the previous illustration shows, when NetCache sends the request to the RADIUS server, NetCache inserts “NetCache” as the user name and the password that is configured in the IP Authentication Password option on the appliance. NetCache also sends Vendor-Netapp-2 and the client IP address.

The RADIUS server response can take a variety of forms, as the following table shows.

If the response is...	The response contains...
Access-reject:	Vendor-Netapp-3 (<i>reject-type</i>)
Access-accept:	The response must contain ◆ Session ID ◆ User name The response can contain either of the following also: ◆ Vendor-Netapp-4 (new host name) ◆ Vendor-Netapp-5 (new URL) If RTMON is used in conjunction with RADIUS extensions, the application server can use the session ID to share information with the RADIUS server.

RADIUS server requirements

Configure the same password on the RADIUS server that is configured in the IP Authentication Password option on the appliance. This password enables the RADIUS server to authenticate to the appliance. (To access the IP Authentication Password option, select on the Setup tab > Authentication > RADIUS > Extensions tab in the Appliance Manager.)

You must program the RADIUS server as follows:

- ◆ The RADIUS server must recognize the Vendor-Netapp-2 attribute and send the Vendor-Netapp-3 attribute.
- ◆ If you are using the Send URL option on the RADIUS page in the Appliance Manager, the RADIUS server must recognize the Vendor-Netapp-1 attribute.
- ◆ If you want the RADIUS server to rewrite client requests to alternative contents, the RADIUS server must be able to send the Vendor-Netapp-4 or Vendor-Netapp-5 attribute with Access-Accept responses.

Standard RADIUS attributes that NetCache sends

About this section

This section describes the RADIUS attributes that NetCache always sends to RADIUS servers, regardless of whether NetApp RADIUS extensions are used.

User-Name

Contained in: Access-Request

Format: String

- ◆ Length: ≥ 3
- ◆ Type: 1 for User-Name

Description: Indicates the name of the user to be authenticated. This attribute must be sent in Access-Request packets if it is available.

See RFC 2865 for more information about this attribute.

User-Password

Contained in: Access-Request

Format: String

- ◆ Length: At least 18 and no longer than 130
- ◆ Type: 2

Description: Indicates the password of the user to be authenticated.

See RFC 2865 for more information about this attribute.

NAS-IP-Address

Contained in: Access-Request

Format:

- ◆ Length: 6
- ◆ Type: 4
- ◆ Address: Four octets

Description: Indicates the identifying IP address of the NAS that is requesting authentication of the user. This attribute should be unique to the NAS within the scope of the RADIUS server. Either the NAS-IP-Address or NAS Identifier must be present in an Access-Request packet.

Calling-Station-Id

Contained in: Access-Request

Format:

- ◆ Length: ≥ 3
- ◆ Type: 31

Description: RFC 2865 indicates that the Calling-Station-ID field can be an arbitrary byte string. NetCache handles the Calling-Station-ID as follows:

If IP authentication...	Then...
<i>Is</i> configured on the NetCache appliance	NetCache sends the IP address as a string in ASCII text.
<i>Is not</i> configured on the NetCache appliance	NetCache sends the IP address in its binary form in network byte order.

NetApp vendor-specific attributes

About vendor-specific attributes

Vendor-specific attributes enable vendors to support their own extended attributes that are not suitable for general RADIUS use. The dictionary files of the RADIUS servers that will use NetApp’s vendor-specific attributes must be configured with NetApp’s set of vendor-specific attributes. If the RADIUS server receives a vendor-specific attribute from the NetCache appliance but has no entry in its dictionary file, the server ignores the data.

Types common to all NetApp RADIUS extensions are

- ◆ Vendor-specific attribute type: 26
- ◆ NetApp vendor ID: 789

Packet requirements

The following table shows requirements for Access-Accept and Access-Reject packets.

Packet type	Requirements
Access-Accept	◆ They must contain a user-name attribute if IP authentication is enabled. ◆ They can contain vendor-specific attributes, for example, a request URL. The RADIUS server can send a new host name or a new URL to NetCache. (The RADIUS server knows what the client has requested.) However, if IP authentication is not enabled on the appliance, the appliance ignores the new host name or new URL.
Access-Reject	The following requirements apply only when IP authentication is enabled on the appliance. That is, if the Send URL option is enabled on the appliance without IP authentication, these requirements do not apply. ◆ They must contain a vendor-specific reject type that maps to a NetApp vendor-specific reject-type. Note: This requirement does not conform to RFC 2865. ◆ They must contain a reply message. The reply message varies according to the streaming media protocol. For RTSP, NetCache can pass on a text message from the RADIUS server that describes the problem to the user. For MMS, the user sees a generic Windows Media error message.

Vendor-Netapp-1

Contained in: Access-Request

Format: String

Vendor-specific-attribute type: 1

Description: Contains the complete requested URL. The maximum length of the URL that NetCache sends depends on the remaining space in the packet after NetCache inserts the following into the packet:

- ◆ User-Name
- ◆ User-Password
- ◆ NAS-IP-Address
- ◆ Calling-Station-ID
- ◆ Vendor-Netapp.2

The current total packet length is 1,000 bytes.

If the Send URL option on the NetCache appliance Authentication - RADIUS page is configured, the RADIUS server must recognize Vendor-Netapp-1 if it is to process the URL. (To access the Send URL option, select Setup tab > Authentication > RADIUS > Extensions tab.)

Vendor-Netapp-2

Contained in: Access-Request

Format: Integer

Vendor-specific-attribute type: 2

Possible values: 1 or 2. A value of 1 is for the IP address. A value of 2 is for user/password authentication.

Description: Describes the contents of the User-Name and User-Password fields.

If NetCache sends to the RADIUS server...	Then...
A value of 1 for Vendor-Netapp-2	The User-Name is set to “NetCache” and the User-Password is set to an encrypted representation of a configured NetCache password (IP Authentication Password or, in the CLI, config.auth.radius.ip_authenticate.pass). Authentication will then be performed on the basis of Calling-Station-ID.
A value of 2 for Vendor-Netapp-2	Per-user information is contained in the User-Name and User-password encrypted fields, which will be used for authentication. (The fields are encrypted according to RFC 2865.)

The client IP address stored in the Calling-Station-ID will be encoded as a dotted decimal string, for example, 10.10.10.1.

Vendor-Netapp-3

Contained in: Access-Reject

Format: Integer

Vendor-specific-attribute type: 3

Possible values: 1, 2, or 3

Description: Action that NetCache takes based on the value returned by the RADIUS server.

Value	RADIUS server result	NetCache determination of action to take based on value
1	Authentication failed	Prompt for user name and password if prompting has not already occurred
2	Authorization failed	Send a Reply-Message to the client

Value	RADIUS server result	NetCache determination of action to take based on value
3	Other (not categorized by a value of 1 or 2)	Deny the request

Vendor-Netapp-4

Contained in: Access-Accept

Format: String

Vendor-specific-attribute type: 4

Description: Contains the new origin server host name or IP address in dotted decimal notation.

If NetCache receives this attribute from the RADIUS server, NetCache rewrites the host name portion of the original URL to use the host name that was sent. If the RADIUS server does not send this attribute, NetCache does nothing.

If NetCache is configured for IP authentication, NetCache will look for User-Name, Class, and either the Vendor-Netapp-4 or Vendor-Netapp-5 attribute. The RADIUS server can send either Vendor-Netapp-4 or Vendor-Netapp-5, but not both.

If you are using RADIUS extensions for access control in a scenario such as in [“Scenario: Using the IP authentication feature for access control by URL”](#) on page 36, the RADIUS server should send Vendor-Netapp-4 or Vendor-Netapp-5.

Vendor-Netapp-5

Contained in: Access-Accept

Format: String

Vendor-specific-attribute type: 5

Description: Contains the complete URL in the packet.

If NetCache receives this attribute from the RADIUS server, NetCache rewrites the URL to the URL that was sent. If the RADIUS server does not send this attribute, NetCache does nothing.

If you are using RADIUS extensions for access control in a scenario such as in “[Scenario: Using the IP authentication feature for access control by URL](#)” on page 36, the RADIUS server should send Vendor-Netapp-4 or Vendor-Netapp-5.

Index

A

- access control with IP authentication 36
- Access-Accept packets, requirements 47
- Access-Reject packets, requirements 47
- accounting in e-Commerce
 - defined 5
 - protocols that provide it 6
 - with RADIUS extensions 11, 37
 - scenario 35
- API
 - RADIUS extensions 41
 - RTMON 21
- application server groups
 - considerations for 16
 - data each server receives 15
 - described 14
 - purpose of 14
 - what you must configure 14
- application servers
 - defined 2
 - IP authentication and RADIUS server load
 - issues 30
 - number with RTMON 16
 - programming for RADIUS extensions 41
 - programming for RTMON 19, 21
- attributes
 - RADIUS
 - Calling-Station-Id 46
 - NAS-IP-Address 45
 - User-Name 45
 - User-Password 45
 - Vendor-Netapp-1 48
 - Vendor-Netapp-2 48
 - Vendor-Netapp-3 49
 - Vendor-Netapp-4 50
 - Vendor-Netapp-5 50
 - RTMON, user 18
- authentication
 - defined 5
 - promptless 10
 - RADIUS extensions described 11
 - standard methods described 12

- authentication components for e-Commerce
 - IP authentication
 - about 10, 30
 - versus standard RADIUS 30
 - protocols available with 6
 - RADIUS extensions
 - compared to standard RADIUS 30
 - described 10
 - functions with 11
 - planning for 29
 - standard methods, described 12
- authorization
 - defined 5
 - protocols that provide 6

B

- billing applications
 - how they use RTMON data 7
 - working with NetCache 2

C

- Calling-Station-Id attribute, defined 46

D

- data to application servers with RTMON 17
- Digital Rights Management
 - integration with NetCache 4

E

- e-Commerce technologies
 - authentication components 10
 - RADIUS extensions 10, 29
 - standard RADIUS 12
 - benefits of NetCache solutions 3
 - determining requirements for 5
 - monitoring component described 7, 13
 - overview of NetCache features 2
 - protocols available 6
 - control components 5, 6

monitoring components 5, 6
enterprises, e-Commerce technologies for 2

F

failover with RTMON 15
firewalls with RADIUS extensions 11

H

Heartbeat events with RTMON
described 18
example 28

I

IP authentication
configuration options on appliance 32
defined 10, 30
illustrated 10, 33, 36
load on RADIUS servers 30
process described 31
protocol supported 30
scenario
access control 36
user name validation 33
versus standard RADIUS 30

L

load balancing with RTMON 15

M

monitoring servers. *See* application servers,
RTMON

N

NAS-IP-Address attribute, defined 45
NAT with RADIUS extensions 11
NetApp RADIUS extensions 6
features it provides 6
functions with 11
NetApp vendor ID 47
NetApp vendor-specific attribute type 47

P

Pay-Per-View, NetCache features for 2
promptless authentication
with RADIUS extensions 10
protocols
in NetCache technology suite 6
RADIUS with extensions
application for e-Commerce 6
RADIUS without extensions
application for e-Commerce 6
RTMON 6
standard authentication
application for e-Commerce 6

R

RADIUS
standard attributes NetCache sends 45
standard transaction illustrated 42
vendor-specific attributes. *See* RADIUS
extensions, NetApp RADIUS
extensions

RADIUS attributes (standard)

Calling-Station-Id 46
NAS-IP-Address 45
User-Name 45
User-Password 45

RADIUS extensions 6

API 41
application servers, programming 41
configuration options on appliance 31, 32
described 10, 29
features it provides 6
firewalls with 11
functions with 11
IP authentication versus standard RADIUS 30
load on RADIUS servers 30
NAT with 11
planning for 29
protocol supported 30
RADIUS server requirements 44
scenario
IP authentication for access control 36
user name validation 33
transaction illustrated 43

- vendor-specific attributes defined 47
- RADIUS servers
 - programming requirements for RADIUS extensions 44
 - traffic with IP authentication 30
- Real-Time Monitoring protocol. *See* RTMON
- redundancy with RTMON 19
- RTMON
 - API 21
 - application server groups 14
 - considerations for 16
 - described 14
 - purpose of 14
 - what you must configure 14
 - application servers, programming 19, 21
 - buffering and failover 15
 - communication with server 8
 - data sent to application servers 8, 17
 - how servers use 7
 - failover with 15
 - features it provides 6, 8
 - Heartbeat events described 18, 28
 - load balancing with 15
 - monitoring illustrated 7
 - number of application servers with 16
 - planning for 13, 14
 - redundancy 19
 - schema 9
 - status events 18
 - described 28
 - example 28
 - TCP Socket with, avoiding 16
 - user attributes
 - available 18
 - defined 17
 - descriptions 24
 - user events
 - available 17
 - defined 17
 - descriptions 23

S

- service providers
 - e-Commerce technologies for 2
- SOAP with RTMON 9
- status events with RTMON 18, 28
 - example 28

T

- TCP Socket with RTMON 16

U

- user attributes (RTMON)
 - available 18
 - defined 17
 - descriptions 24
- user events (RTMON)
 - available 17
 - defined 17
 - descriptions 23
- User-Name attribute, defined 45
- User-Password attribute, defined 45

V

- Vendor-Netapp-1 attribute, defined 48
- Vendor-Netapp-2 attribute, defined 48
- Vendor-Netapp-3 attribute, defined 49
- Vendor-Netapp-4 attribute, defined 50
- Vendor-Netapp-5 attribute, defined 50

X

XML

- API for RTMON 21
- schema with RTMON 9

