CHAPTER 9

Configuring Stickiness

This chapter describes how to configure stickiness on the Cisco Application Control Engine (ACE) using Cisco Application Networking Manager (ANM).

Note When naming ACE objects (such as a real server, virtual server, parameter map, class map, health probe, and so on), enter an alphanumeric string of 1 to 64 characters, which can include the following special characters: underscore (_), hyphen (-), and dot (.). Spaces are not allowed.

If you are using ANM with an ACE module or ACE appliance and you configure a named object at the ACE CLI, keep in mind that ANM does not support all of the special characters that the ACE CLI allows you to use when configuring a named object. If you use special characters that ANM does not support, you may not be able to import or manage the ACE using ANM.

This chapter includes the following sections:

Information About Stickiness

When customers visit an e-commerce site, they usually start out browsing the site. The site may require that the client become “stuck” to one server once the connection is established, or once client starts to build a shopping cart.

In either case, once the client adds items to the shopping cart, it is important that all of the client requests get directed to the same server so that all the items are contained in one shopping cart on one server. An instance of a customer’s shopping cart is typically local to a particular web server and is not duplicated across multiple servers.

E-commerce applications are not the only types of applications that require stickiness. Any web application that maintains client information may require stickiness, such as banking applications or online trading. Other uses include FTP and HTTP file transfers.

Stickiness allows the same client to maintain multiple simultaneous or subsequent TCP or IP connections with the same real server for the duration of a session. A session is series of transactions between a client and a server over some finite period of time (from several minutes to several hours). This feature is particularly useful for e-commerce applications where a client needs to maintain multiple connections with the same server while shopping online, especially while building a shopping cart and during the checkout process.

Depending on the configured SLB policy, the ACE sticks a client to an appropriate server after the ACE has determined which load-balancing method to use. If the ACE determines that a client is already stuck to a particular server, then the ACE sends that client request to that server, regardless of the load-balancing criteria specified by the matched policy. If the ACE determines that the client is not stuck to a particular server, it applies the normal load-balancing rules to the content request.

You can configure stickiness to stick a client to a real server that is associated with a server farm or you can use the *buddy* sticky group feature to enable persistence to a real server or real server group across multiple server farms (see the "Buddy Sticky Groups" section on page 9-6).

For information about stickiness, see the following topics:

Related Topics

- Configuring Virtual Server Default Layer 7 Load Balancing, page 7-50
- Configuring Sticky Groups, page 9-12

Sticky Types

All ACE devices support stickiness based on the following:

- HTTP cookies
- HTTP headers
- IP addresses
- HTTP content
- IP Netmask
- IPv6 Prefix
- Layer 4 payloads
- RADIUS attributes
- RTSP headers
- SIP headers
- SSL session ID

This section includes the following topics:

- HTTP Header Stickiness, page 9-4
- IP Netmask and IPv6 Prefix Stickiness, page 9-4
- Layer 4 Payload Stickiness, page 9-4
- RADIUS Stickiness, page 9-5
- RTSP Header Stickiness, page 9-5
- SIP Header Stickiness, page 9-5
- SSL Stickiness, page 9-6

HTTP Content Stickiness

HTTP content stickiness allows you to stick a client to a server based on the content of an HTTP packet. You can specify a beginning pattern and ending pattern, the number of bytes to parse, and an offset that specifies how many bytes to ignore from the beginning of the data.

Related Topics

- Configuring Stickiness, page 9-1
- Sticky Types, page 9-2
- Sticky Groups, page 9-6
- Sticky Table, page 9-11

HTTP Cookie Stickiness

Client *cookies* uniquely identify clients to the ACE and the servers that provide content. A cookie is a small data structure within the HTTP header that is used by a server to deliver data to a web client and request that the client store the information. In certain applications, the client returns the information to the server to maintain the connection state or persistence between the client and the server.

When the ACE examines a request for content and determines through policy matching that the content is sticky, it examines any cookie or URL present in the content request. The ACE uses the information in the cookie or URL to direct the content request to the appropriate server.

The ACE supports the following types of cookie stickiness:

- Dynamic cookie learning

  You can configure the ACE to look for a specific cookie name and automatically learn its value either from the client request HTTP header or from the server Set-Cookie message in the server response. Dynamic cookie learning is useful when dealing with applications that store more than just the session ID or user ID within the same cookie. Only very specific bytes of the cookie value are relevant to stickiness.

  By default, the ACE learns the entire cookie value. You can optionally specify an offset and length to instruct the ACE to learn only a portion of the cookie value.

  Alternatively, you can specify a secondary cookie value that appears in the URL string in the HTTP request. This option instructs the ACE to search for (and eventually learn or stick to) the cookie information as part of the URL. URL learning is useful with applications that insert cookie information as part of the HTTP URL. In some cases, you can use this feature to work around clients that reject cookies.

- Cookie insert

  The ACE inserts the cookie on behalf of the server upon the return request, so that the ACE can perform cookie stickiness even when the servers are not configured to set cookies. The cookie contains information that the ACE uses to ensure persistence to a specific real server.

Related Topics

- Configuring Stickiness, page 9-1
- Sticky Types, page 9-2
- Sticky Groups, page 9-6
- Sticky Table, page 9-11

HTTP Header Stickiness

You can use HTTP-header information to provide stickiness. With HTTP header stickiness, you can specify a header offset to provide stickiness based on a unique portion of the HTTP header.

Related Topics

- Configuring Stickiness, page 9-1
- Sticky Types, page 9-2
- Sticky Groups, page 9-6
- Sticky Table, page 9-11

IP Netmask and IPv6 Prefix Stickiness

You can use the source IP address, the destination IP address, or both to uniquely identify individual clients and their requests for stickiness purposes based on their IP netmask or IPv6 prefix. However, if an enterprise or a service provider uses a megaproxy to establish client connections to the Internet, the source IP address no longer is a reliable indicator of the true source of the request. In this case, you can use cookies or one of the other sticky methods to ensure session persistence.

Note IPv6 requires ACE module and ACE appliance software Version A5(1.0) or later.

Related Topics

- Configuring Stickiness, page 9-1
- Sticky Types, page 9-2
- Sticky Groups, page 9-6
- Sticky Table, page 9-11

Layer 4 Payload Stickiness

Layer 4 payload stickiness allows you to stick a client to a server based on the data in Layer 4 frames. You can specify a beginning pattern and ending pattern, the number of bytes to parse, and an offset that specifies how many bytes to ignore from the beginning of the data.

Related Topics

- Configuring Stickiness, page 9-1
- Sticky Types, page 9-2
- Sticky Groups, page 9-6
- Sticky Table, page 9-11

RADIUS Stickiness

RADIUS stickiness can be based on the following RADIUS attributes:

- Calling Station ID
- Username

Related Topics

- Configuring Stickiness, page 9-1
- Sticky Types, page 9-2
- Sticky Groups, page 9-6
- Sticky Table, page 9-11

RTSP Header Stickiness

Real time streaming protocol (RTSP) stickiness is based on information in the RTSP session header. With RTSP header stickiness, you can specify a header offset to provide stickiness based on a unique portion of the RTSP header.

Related Topics

- Configuring Stickiness, page 9-1
- Sticky Types, page 9-2
- Sticky Groups, page 9-6
- Sticky Table, page 9-11

SIP Header Stickiness

Session initiation protocol (SIP) header stickiness is based on the SIP Call-ID header field. SIP header stickiness requires the entire SIP header, so you cannot specify an offset.

Related Topics

- Configuring Stickiness, page 9-1
- Sticky Types, page 9-2
- Sticky Groups, page 9-6
- Sticky Table, page 9-11

SSL Stickiness

Note This feature requires ACE software Version A5(2.0) or later.

SSL stickiness allows you to stick a client to a server based on the SSL session ID. You can associate an SSL sticky group with an HTTPS server load balancing policy map.

Related Topics

- Configuring Stickiness, page 9-1
- Sticky Types, page 9-2
- Sticky Groups, page 9-6
- Sticky Table, page 9-11

Sticky Groups

Sticky groups allow the ACE to keep a client stuck to a real server or group of real servers within a server farm. The ACE uses the concept of sticky groups to configure stickiness. A sticky group allows you to specify sticky attributes. After you configure a sticky group and its attributes, you associate the sticky group with a Layer 7 policy-map action in a Layer 7 server load balancing (SLB) policy map.You can create a maximum of 4096 sticky groups in each context. Each sticky group that you configure on the ACE contains a series of parameters that determine the following:

- Sticky method
- Timeout
- Replication
- Sticky method-specific attributes

Note The context in which you configure a sticky group must be associated with a resource class that allocates a portion of ACE resources to stickiness. See the "Using Resource Classes" section on page 6-44 for information about configuring ACE resources.

Related Topics

- Configuring Stickiness, page 9-1
- Sticky Types, page 9-2
- Sticky Table, page 9-11

Buddy Sticky Groups

Note The buddy sticky group feature requires ACE software Version A5(2.0) or later.

Buddy sticky groups allow the ACE to keep a client stuck to a real server or group of real servers even when the client requests are processed by different server farms.

To use the buddy sticky group feature, you perform the following steps:

1. Create real server buddy groups when specifying the real servers in a server farm (see the "Configuring Server Farms" section on page 8-31).
2. Create sticky server farm buddy groups when specifying the server farms in a sticky group (see the "Configuring Sticky Groups" section on page 9-12). You make each sticky server farm to be buddied together a group *member*.

This section describes the following buddy sticky group applications:

- One-to-one association—Sticks the client to the same physical server instances in two different server farms.
- Asymmetric association—Sticks a client to a real server that is configured across different serverfarms even when the client comes back with a non-HTTP request or different HTTP header.
- Many-to-one association—Sticks multiple, first-tier real servers to one real server in a second tier that contains fewer servers.

This section includes the following topics:

Guidelines and Restrictions

Observe the following guidelines and restrictions when using the buddy sticky group feature:

- When two sticky groups with different timeout values are buddied together, the ACE uses the shortest timeout value for the buddy group.
- Sticky groups that are buddied together must of the same type, such as all IP-sticky, all http-cookie, and so forth. The ACE does not support different types of sticky groups buddied together.
- When two sticky groups are buddied together and one of them is configured for timeout active connections, the member group is also configured for timeout active connections.
- When two sticky groups are configured with different IP netmask (IPv4) or prefix-length (IPv6), the ACE uses the one with the most granular netmask or prefix-length.
- When a static entry is created under a buddy sticky group, its behavior is unchanged and it sticks to the same real server configured regardless of the buddy group that real server is associated with.
- Before you can configure a sticky group as a member, you must have a server farm configured under that sticky group and all the real servers that belong to that server farm have buddy group configured under them. This requirement prevents invalid configurations.
- The ACE does not support configuring the following types of sticky groups as buddy sticky group members:
  - SSL
  - RTSP Header
- The ACE supports PTMP sticky group such as SIP sticky; however, you must make sure that the configuration is the same across both sticky groups for the buddy sticky group feature to work.

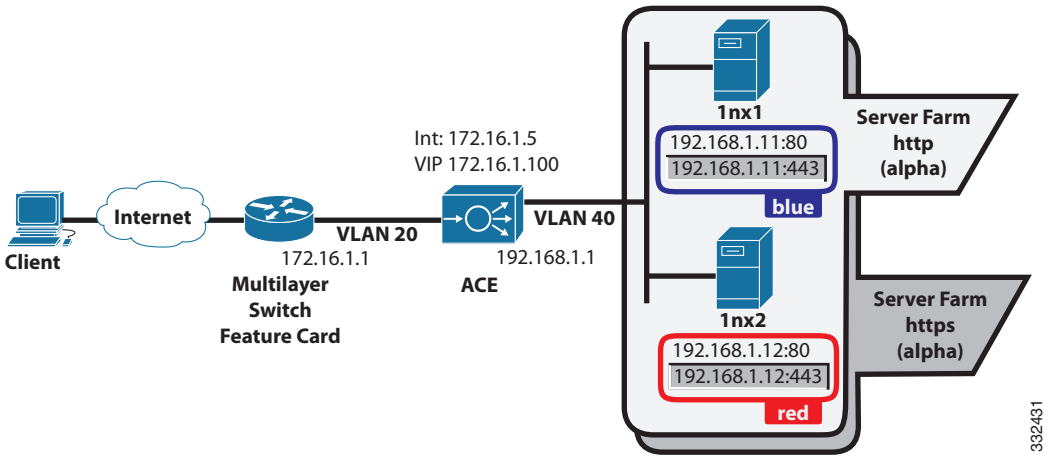
- For real server backup applications:
  - We recommend only one level of backup-rserver with buddy sticky.
  - If you add a buddy group to the primary real server, the backup server inherits this buddy group. However, if you remove the buddy group from the primary real server, the buddy group is not removed from the backup real server and vice versa.

One-to-One Association Example

In a one-to-one buddy sticky group association, you create a buddy sticky group that sticks a client to the same physical server instances in two different server farms. In the network example shown in Figure 9-1, the ACE is configured with the following server farms, their associated real servers, and the buddy sticky groups that group both items:

Server Farm	Server Farm Buddy Member Group	Real Server	Real Server Buddy Group
http (for HTTP requests)	alpha	1nx1:192.168.1.11:80	blue
		1nx2:192.168.1.12:80	red
https (for HTTPS requests)	alpha	1nx1:192.168.1.11:443	blue
		1nx2:192.168.1.12:443	red

Figure 9-1 Buddy Sticky Groups: One-to-One Association



The ACE is configured to load balance HTTP requests to server farm http using either real server 1nx1:192.168.1.11:80 or 1nx2:192.168.1.12:80. The ACE is also configured to load balance HTTPS requests using server farm https and either real server 1nx1:192.168.1.11:443 or 1nx2:192.168.1.12:443. The buddy groups allow the ACE to stick a client to the same real server (for example, 1nx1) while building a shopping cart using HTTP requests and then checking out using HTTPS.

In this example, the client hits VIP 172.16.1.100, destination port 80 with an HTTP request to begin to build a shopping cart. The ACE load balances the request to server farm http, real server 1nx1:192.168.1.11:80 and creates a sticky entry based on the corresponding sticky group (for example, source IP address) that sticks the client to the real server while the client builds their shopping cart. When the client moves to the secured connection (port 443) for checkout, it hits the VIP with destination port

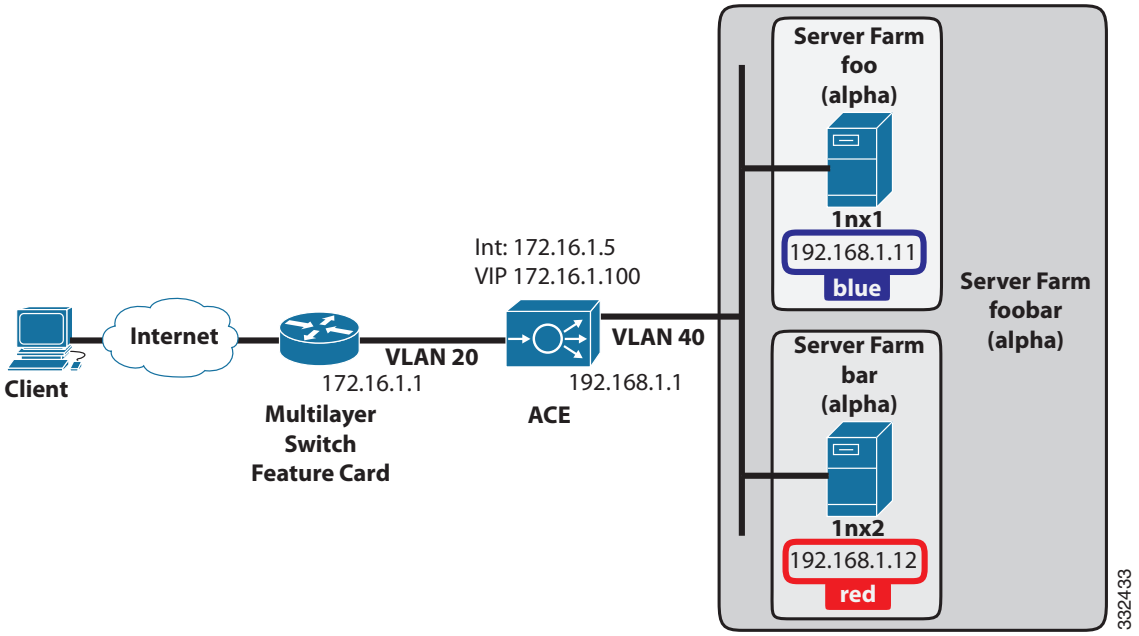
443 and the ACE sends the client to server farm https. The ACE finds an existing sticky entry with real server Inx1:192.168.1.11:80 and directs the client to 1nx1:192.168.1.11:443 because the two real servers are buddied together under the blue buddy group.

Asymmetric Association Example

In an asymmetric buddy sticky group association, you create a buddy sticky group that sticks all Layer 7 traffic from a client to a specific real server even when some of the traffic does not match the Layer 7 class map. In the network example shown in Figure 9-2, the ACE is configured to include the following server farms, their associated real servers, and real server buddy sticky groups.

Server Farm	Server Farm Buddy Member Group	Real Server	Real Server Buddy Group
foo bar	alpha	1nx1	blue
		1nx2	red
foo	alpha	1nx1	blue
bar	alpha	1nx2	red

Figure 9-2 Buddy Sticky Groups: Asymmetric Association



The ACE is configured to send client traffic with Layer 3 matches to server farm foobar, which contains the nested server farms foo and bar. The ACE load balances the client traffic to one of the nested server farms based on Layer 7 class map matches. By defining buddy sticky groups, the ACE is also able to stick non-matching client traffic to the same real server.

In this example, the client sends traffic with Layer 3 matches that the ACE directs and sticks (using ip sticky) to server farm foobar. The ACE uses a Layer 7 class map to check for HTTP URL and if present, sends the traffic to server farm foo and sticks the client traffic to that server using sticky that is based on

the source IP address. Using a buddy stick group, the ACE uses the sticky entry to send any other traffic type from the client to the same real server. For example, if the ACE sticks the client HTTP traffic to server farm foo:real server lnx1 based on a Layer 7 class map match, the buddy stick group allows the ACE to send non-HTTP traffic from the client to the same real server.

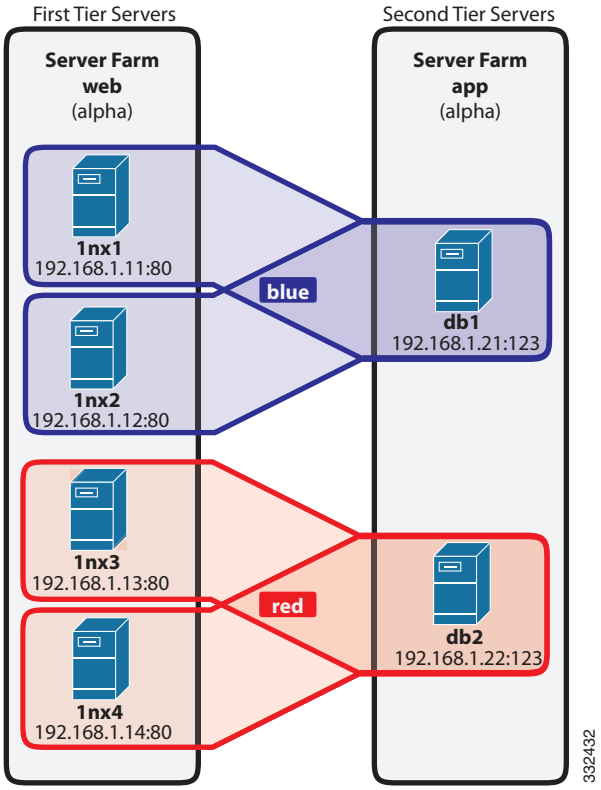
Many-to-One Association Example

In a many-to-one buddy sticky group association, you create a buddy sticky group that sticks a group of real servers to a specific real server, which is useful when clients are load balanced to a first-tier server farm containing many real servers and are then directed to a second tier server farm that contains fewer real servers. In this type of application, you create buddy sticky groups that stick each first-tier real server group to a specific second-tier real server.

In the network example shown in Figure 9-3, the ACE is configured with the following server farms, their associated real servers, and assigned real server buddy groups:

Server Farm	Server Farm Buddy Member Group	Real Server	Real Server Buddy Group
web (first tier)	alpha	1nx1:192.168.1.11:80	blue
		1nx2:192.168.1.12:80	blue
		1nx3:192.168.1.13:80	red
		1nx4:192.168.1.14:80	red
app (second tier)	alpha	db1:192.168.1.21:123	blue
		db1:192.168.1.22:123	red

Figure 9-3 Buddy Sticky Groups: Many-to-One Association



The buddy sticky groups blue and red divide the first-tier real servers into groups and then sticks each of these groups to a specific second-tier real server.

In this example, when the ACE load balances clients to either real server 1nx1 or 1nx2 in the server farm web, the clients are directed only to real server db1 when they are ready to move to the server farm app. Notice also that clients that the ACE load balances to 1nx3 and 1nx4 are directed only to real server db2 when they are ready to move to the server farm app.

Sticky Table

The ACE uses a sticky table to keep track of sticky connections. Table entries are as follows:

- Sticky groups
- Sticky methods
- Sticky connections
- Real servers

The sticky table can hold a maximum of four million entries (four million simultaneous users). When the table reaches the maximum number of entries, additional sticky connections cause the table to wrap and the first users become unstuck from their respective servers.

The ACE uses a configurable timeout mechanism to age out sticky table entries. When an entry times out, it becomes eligible for reuse. High connection rates may cause the premature aging out of sticky entries. In this case, the ACE reuses the entries that are closest to expiration first.

Sticky entries can be either dynamic (generated by the ACE on demand) or static (user-configured). When you create a static sticky entry, the ACE places the entry in the sticky table immediately. Static entries remain in the sticky database until you remove them from the configuration. You can create a maximum of 4096 static sticky entries in each context.

If the ACE takes a real server out of service for whatever reason (probe failure, no inservice command, or ARP timeout), the ACE removes from the database any sticky entries that are related to that server.

Related Topics

- Configuring Stickiness, page 9-1
- Sticky Types, page 9-2
- Sticky Groups, page 9-6

Configuring Sticky Groups

You can configure sticky groups. Stickiness (or session persistence) is a feature that allows the same client to maintain multiple simultaneous or subsequent TCP connections with the same real server for the duration of a session. A session is a series of transactions between a client and a server over some finite period of time (from several minutes to several hours). This feature is particularly useful for e-commerce applications where a client needs to maintain multiple TCP connections with the same server while shopping online, especially while building a shopping cart and during the checkout process.

E-commerce applications are not the only types of applications that require stickiness. Any web application that maintains client information may require stickiness, such as banking applications or online trading. Other uses include FTP and HTTP file transfers.

The ACE uses the concept of sticky groups to configure stickiness. A sticky group allows you to specify sticky attributes. After you configure a sticky group and its attributes, you associate the sticky group with a Layer 7 policy-map action in a Layer 7 SLB policy map.

Note (Pre ACE version A4(1.0) module or appliance only) The context in which you configure a sticky group must be associated with a resource class that allocates a portion of ACE resources to stickiness. See the "Using Resource Classes" section on page 6-44 for information about configuring ACE resources.

Assumption

(Pre ACE version A4(1.0) module or appliance only) The context in which you are configuring a sticky group is associated with a resource class that allocates resources to stickiness.

Procedure

Step 1 Choose **Config > Devices >** *context* **> Load Balancing > Stickiness**.

The Sticky Groups table appears.

Step 2 In the Sticky Groups table, click **Add** to add a new sticky group, or choose an existing sticky group that you want to modify and click **Edit**.

Step 3 Configure the sticky group using the information in Table 9-1.

Note Fields and information related to IPv6 require ACE module and ACE appliance software Version A5(1.0) or later.

Table 9-1 Sticky Group Attributes

Field	Description
Group Name	Sticky group identifier. Valid entries are unquoted text strings with no spaces and a maximum of 64 alphanumeric characters.
Type	Method to be used when establishing sticky connections and to configure any type-specific attributes. The choices are as follows: • **HTTP Content**—The ACE sticks client connections to the same real server based on a string in the data portion of the HTTP packet. See Table 9-2 for additional configuration options. • **HTTP Cookie**—The ACE either learns a cookie from the HTTP header of a client request or inserts a cookie in the Set-Cookie header of the response from the server to the client and then uses the learned cookie to provide stickiness between the client and server for the duration of the transaction. See Table 9-3 for additional configuration options. • **HTTP Header**—The ACE sticks client connections to the same real server based on HTTP headers. See Table 9-4 for additional configuration options. • **IP Netmask**—The ACE sticks a client to the same server for multiple subsequent connections as needed to complete a transaction using the client source IPv4 IP address, the destination IPv4 IP address, or both. You can optionally configure an IPv6 prefix length with this sticky type. IPv6 requires ACE module and ACE appliance software Version A5(1.0) or later. See Table 9-5 for additional configuration options. **Note** If an organization uses a megaproxy to load balance client requests across multiple proxy servers when a client connects to the Internet, the source IP address is no longer a reliable indicator of the true source of the request. In this situation, you can use cookies or another sticky method to ensure session persistence. • **V6 Prefix**—(Option that appears only for ACE module and ACE appliance software Version A5(1.0) or later.) The ACE appliance sticks a client to the same server for multiple subsequent connections as needed to complete a transaction using the client source IP address, the destination IP address, or both based on their IPv6 prefix. You can optionally configure an IPv4 netmask with this sticky type. See Table 9-6 for additional configuration options. • **Layer 4 Payload**—The ACE sticks client connections to the same real server based on a string in the payload portion of the Layer 4 protocol packet. See Table 9-7 for additional configuration options. • **RADIUS**—The ACE sticks client connections to the same real server based on a RADIUS attribute. See Table 9-8 for additional configuration options. • **RTSP Header**—The ACE sticks client connections to the same real server based on the RTSP Session header field. See Table 9-9 for additional configuration options. • **SIP Header**—The ACE sticks client connections to the same real server based on the SIP Call-ID header field. • **SSL**—The ACE sticks client connections to the same real server based on the SSL session ID. **Note** This option requires ACE software Version A5(2.0) or later. **Note** This option is not available with the ACE NPE software version (see the "Information About the ACE No Payload Encryption Software Version" section on page 1-3).

Table 9-1 Sticky Group Attributes (continued)

Field	Description
Cookie Name	This option appears for sticky type HTTP Cookie. Enter a unique identifier for the cookie. Valid entries are unquoted text strings with no spaces and a maximum of 64 alphanumeric characters.
Enable Insert	This option appears only for sticky type HTTP Cookie. Check this check box if the ACE appliance is to insert a cookie in the Set-Cookie header of the response from the server to the client. This option is useful when you want to use a session cookie for persistence but the server is not currently setting the appropriate cookie. When selected, the ACE appliance selects a cookie value that identifies the original server from which the client received a response. For subsequent connections of the same transaction, the client uses the cookie to stick to the same server. Clear this check box to disable cookie insertion.
Browser Expire	This option appears for sticky type HTTP Cookie and you select Enable Insert. Check this check box to allow the client's browser to expire a cookie when the session ends. Clear this check box to disable browser expire.
Offset (Bytes)	This option appears for sticky types HTTP Cookie and HTTP Header. Enter the number of bytes the ACE appliance is to ignore starting with the first byte of the cookie. Valid entries are integers from 0 to 999. The default is 0 (zero), which indicates that the ACE appliance does not exclude any portion of the cookie.
Length (Bytes)	This option appears for sticky types HTTP Cookie, HTTP Header, and SSL. Enter the length of the portion of the cookie (starting with the byte after the offset value) that the ACE appliance is to use for sticking the client to the server. For the SSL sticky type, enter the SSL session ID length that needs to be parsed. Valid entries are integers from 1 to 1000.
Secondary Name	This option appears only for sticky type HTTP Cookie. Enter an alternate cookie name that is to appear in the URL string of the Web page on the server. The ACE appliance uses this cookie to maintain a sticky connection between a client and a server and adds a secondary entry in the sticky table. Valid entries are unquoted text strings with no spaces and a maximum of 64 characters.
Header Name	This option appears for sticky type HTTP Header. Select the HTTP header to use for sticking client connections.
Netmask	This option appears only for sticky type IP Netmask. Select the netmask to apply to the source IP address, the destination IP address, or both.
IPv4 Netmask	This option appears only for sticky type IP Netmask or IPv6 Prefix (IPv6 requires ACE module and ACE appliance software Version A5(1.0) or later). This option is mandatory for the sticky type IP Netmask and optional for the sticky type IPv6 Prefix. Select the netmask to apply to the source IP address, the destination IP address, or both.
IPv6 Prefix Length	This option appears only for ACE module and ACE appliance software Version A5(1.0) or later and for sticky type IPv6 Prefix or IP Netmask. This option is mandatory for the sticky type IPv Prefix and optional for the sticky type IP Netmask. Enter the IPv6 prefix length to apply to the source IP address, the destination IP address, or both.

Table 9-1 Sticky Group Attributes (continued)

Field	Description
Address Type	This option appears only for sticky type IP Netmask or IPv6 Prefix (IPv6 requires ACE module and ACE appliance software Version A5(1.0) or later). Indicate whether this sticky type is to be applied to the client source IP address, the destination IP address, or both: • Both—Indicates that this sticky type is to be applied to both the source IP address and the destination IP address. • Destination—Indicates that this sticky type is to be applied to the destination IP address only. Source—Indicates that this sticky type is to be applied to the source IP address only.
Enable Sticky For Response	This check box option appears only for sticky type Layer 4 Payload and requires ACE module software Version A2(1.0) or ACE appliance software Version A3(1.0) or later releases of either software version. Check the check box to instruct the ACE to parse the response bytes from a server and perform sticky learning. Clear the check box when you do not want the ACE to perform this operation.
Sticky Server Farm	Server farm that you want to associate with this sticky group.
Backup Server Farm	Backup server farm that is associated with this sticky group. If the primary server farm is down, the ACE uses the backup server farm.
Aggregate State	Field that appears when a server farm and backup server farm are selected. Check box that indicates that the state of the backup server farm is tied to the virtual server state. Uncheck this check box if the backup server farm is not tied to the virtual server state.
Sticky Enabled On Backup Server Farm	Field that appears when a server farm and backup server farm are selected. Check box that indicates that the backup server farm is sticky. Uncheck this check box if the backup server farm is not sticky.
Buddy Group	This field appears when a server farm is selected and requires ACE software Version A5(2.0) or later. Associate the server farm with an existing buddy sticky group or create a buddy sticky group. When you associate multiple server farms with the same buddy group, client requests are stuck to the same real server even when the requests are processed by different server farms. For more information, see the "Buddy Sticky Groups" section on page 9-6.
Replicate On HA Peer	Check box that indicates that the ACE to replicate sticky table entries on the standby ACE. If a failover occurs and this option is selected, the new active ACE can maintain the existing sticky connections. Uncheck this check box to indicate that the ACE is not to replicate sticky table entries on the standby ACE.
Timeout (Minutes)	Number of minutes that the ACE keeps the sticky information for a client connection in the sticky table after the latest client connection terminates. Valid entries are from 1 to 65535; the default is 1440 minutes (24 hours).
Timeout Active Connections	Check box that specifies that the ACE is to time out sticky table entries even if active connections exist after the sticky timer expires. Uncheck this check box to specify that the ACE is not to time out sticky table entries even if active connections exist after the sticky timer expires. This behavior is the default.

Step 4 Do one of the following:

- Click **Deploy Now** to deploy this configuration on the ACE and save your entries to the running-configuration and startup-configuration files. To configure sticky statics, see the "Configuring Sticky Statics" section on page 9-23.
- Click **Cancel** to exit the procedure without saving your entries and to return to the Sticky Groups table.
- Click **Next** to deploy your entries and to configure another sticky group.

Related Topics

- Configuring Sticky Statics, page 9-23
- Configuring Virtual Context Class Maps, page 14-6
- Configuring Virtual Context Policy Maps, page 14-32
- Configuring Real Servers, page 8-5
- Configuring Server Farms, page 8-31

Sticky Group Attribute Tables

This section describes the different sticky group type-specific attributes.

Note There are no specific sticky group type-specific attributes for SIP Header.

This section includes the following topics:

- HTTP Content Sticky Group Attributes, page 9-17
- HTTP Cookie Sticky Group Attributes, page 9-18
- HTTP Header Sticky Group Attributes, page 9-19
- IP Netmask Sticky Group Attributes, page 9-19
- V6 Prefix Sticky Group Attributes, page 9-20
- Layer 4 Payload Sticky Group Attributes, page 9-20
- RADIUS Sticky Group Attributes, page 9-21
- RTSP Header Sticky Group Attributes, page 9-21
- SSL Header Sticky Group Attributes, page 9-22

HTTP Content Sticky Group Attributes

Table 9-2 describes the HTTP content sticky group attributes.

Table 9-2 HTTP Content Sticky Group Attributes

Field	Description
HTTP Content	Check box that instructs the ACE to use the constant portion of HTTP content to make persistent connections to a specific server. Uncheck the check box to identify specific content for stickiness in the Offset, Length, Begin Pattern, and End Pattern fields. HTTP content may change over time with only a portion remaining constant throughout a transaction between the client and a server.
Offset	Number of bytes that the virtual server is to ignore starting with the first byte of the cookie. Valid entries are from 0 to 999. The default is 0 (zero), which indicates that the virtual server does not exclude any portion of the cookie.
Length (Bytes)	Length of the portion of the cookie (starting with the byte after the offset value) that the ACE is to use for sticking the client to the server. Valid entries are from 1 to 1000.
Begin Pattern	Beginning pattern of the HTTP content payload and the pattern string to match before hashing. If you do not specify a beginning pattern, the ACE begins parsing immediately after the offset byte. You cannot configure different beginning and ending patterns for different server farms that are part of the same traffic classification. Valid entries are unquoted text strings with no spaces and a maximum of 255 alphanumeric characters. You can enter a text string with spaces if you enclose the entire string in quotation marks ("). The ACE supports regular expressions for matching string expressions. Table 14-35 lists the supported characters that you can use for matching string expressions.
End Pattern	Pattern that marks the end of hashing. If you do not specify an end pattern or a length, the ACE continues to parse the data until it reaches the end of the field or packet, or until it reaches the maximum body parse length. You cannot configure different beginning and ending patterns for different server farms that are part of the same traffic classification. Valid entries are unquoted text strings with no spaces and a maximum of 255 alphanumeric characters. You can enter a text string with spaces if you enclose the entire string in quotation marks ("). The ACE supports regular expressions for matching string expressions. Table 14-35 lists the supported characters that you can use for matching string expressions.

HTTP Cookie Sticky Group Attributes

Table 9-3 describes the HTTP cookie sticky group attributes.

Table 9-3 HTTP Cookie Sticky Group Attributes

Field	Description
Cookie Name	Unique identifier for the cookie. Valid entries are unquoted text strings with no spaces and a maximum of 64 alphanumeric characters.
Enable Insert	Check box that determines if the virtual server is to insert a cookie in the Set-Cookie header of the response from the server to the client. This option is useful when you want to use a session cookie for persistence but the server is not currently setting the appropriate cookie. When selected, the virtual server selects a cookie value that identifies the original server from which the client received a response. For subsequent connections of the same transaction, the client uses the cookie to stick to the same server. Uncheck the check box to disable cookie insertion.

Table 9-3 HTTP Cookie Sticky Group Attributes (continued)

Field	Description
Offset	Number of bytes that the virtual server is to ignore starting with the first byte of the cookie. Valid entries are from 0 to 999. The default is 0 (zero), which indicates that the virtual server does not exclude any portion of the cookie.
Length (Bytes)	Length of the portion of the cookie (starting with the byte after the offset value) that the ACE is to use for sticking the client to the server. Valid entries are from 1 to 1000.
Secondary Name	Alternate cookie name that is to appear in the URL string of the web page on the server. The virtual server uses this cookie to maintain a sticky connection between a client and a server and adds a secondary entry in the sticky table. Valid entries are unquoted text strings with no spaces and a maximum of 64 characters.

HTTP Header Sticky Group Attributes

Table 9-4 describes the HTTP header sticky group attributes.

Table 9-4 HTTP Header Sticky Group Attributes

Field	Description
Header Name	HTTP header to use for sticking client connections.
Offset	Number of bytes the virtual server is to ignore starting with the first byte of the cookie. Valid entries are from 0 to 999. The default is 0 (zero), which indicates that the virtual server does not exclude any portion of the cookie.
Length (Bytes)	Length of the portion of the cookie (starting with the byte after the offset value) that the ACE is to use for sticking the client to the server. Valid entries are from 1 to 1000.

IP Netmask Sticky Group Attributes

Table 9-5 describes the IP netmask sticky group attributes.

Table 9-5 IP Netmask Sticky Group Attributes

Field	Description
Netmask	Netmask to apply to the source IP address, destination IP address, or both.
IPv6 Prefix Length	(Optional field that requires ACE module and ACE appliance software Version A5(1.0) or later) IPv6 prefix length to apply to the source IP address, destination IP address, or both.
Address Type	Address type that the sticky type is to be applied to as follows: • **Both**—Sticky type is applied to both the source IP address and the destination IP address. • **Destination**—Sticky type is applied to the destination IP address only. • **Source**—Sticky type applied to the source IP address only.

V6 Prefix Sticky Group Attributes

Table 9-5 describes the V6 prefix sticky group attributes, which requires ACE module and ACE appliance software Version A5(1.0) or later.

Table 9-6 IV6 Prefix Sticky Group Attributes

Field	Description
Prefix Length	(Field that requires ACE module and ACE appliance software Version A5(1.0) or later) IPv6 prefix length to apply to the source IP address, destination IP address, or both.
IPv4 Netmask	(Optional) Netmask to apply to the source IP address, destination IP address, or both.
Address Type	Address type that the sticky type is to be applied to as follows: • **Both**—Sticky type is applied to both the source IP address and the destination IP address. • **Destination**—Sticky type is applied to the destination IP address only. • **Source**—Sticky type applied to the source IP address only.

Layer 4 Payload Sticky Group Attributes

Table 9-7 describes the Layer 4 payload sticky group attributes.

Table 9-7 Layer 4 Payload Sticky Group Attributes

Field	Description
Offset	Number of bytes the virtual server is to ignore starting with the first byte of the cookie. Valid entries are from 0 to 999. The default is 0 (zero), which indicates that the virtual server does not exclude any portion of the cookie.
Length (Bytes)	Length of the portion of the cookie (starting with the byte after the offset value) that the ACE is to use for sticking the client to the server. Valid entries are from 1 to 1000. The default is 1000.
Begin Pattern	Beginning pattern of the Layer 4 payload and the pattern string to match before hashing. If you do not specify a beginning pattern, the ACE begins parsing immediately after the offset byte. You cannot configure different beginning and ending patterns for different server farms that are part of the same traffic classification. Valid entries are unquoted text strings with no spaces and a maximum of 255 alphanumeric characters. You can enter a text string with spaces provided that you enclose the entire string in quotation marks ("). The ACE supports regular expressions for matching string expressions. Table 14-35 lists the supported characters that you can use for matching string expressions.

Table 9-7 Layer 4 Payload Sticky Group Attributes (continued)

Field	Description
End Pattern	Pattern that marks the end of hashing. If you do not specify an end pattern or a length, the ACE continues to parse the data until it reaches the end of the field or packet, or until it reaches the maximum body parse length. You cannot configure different beginning and ending patterns for different server farms that are part of the same traffic classification. Valid entries are unquoted text strings with no spaces and a maximum of 255 alphanumeric characters. You can enter a text string with spaces provided that you enclose the entire string in quotation marks ("). The ACE supports regular expressions for matching string expressions. Table 14-35 lists the supported characters that you can use for matching string expressions.
Enable Sticky For Response	Check box that enables the ACE to parse server responses and perform sticky learning. The ACE uses a hash of the server response bytes to populate the sticky database. The next time that the ACE receives a client request with those same bytes, it sticks the client to the same server. Uncheck the check box to reset the behavior of the ACE to the default of not parsing server responses and performing sticky learning.

RADIUS Sticky Group Attributes

Table 9-8 describes the RADIUS sticky group attributes.

Table 9-8 RADIUS Sticky Group Attributes

Field	Description
RADIUS Types	Choose the RADIUS attribute to use for sticking client connections: • **N/A**—This option is not configured. • **RADIUS Calling ID**—Stickiness is based on the RADIUS framed IP attribute and the calling station ID attribute. • **RADIUS User Name**—Stickiness is based on the RADIUS framed IP attribute and the username attribute.
Enter User IPv6Prefix Length[1]	Enter the IPv6 prefix length for IPv6 end user packets when using RADIUS IPv6 attributes. For RADIUS-framed IP sticky using IPv6, the sticky entry is based on the framed IPv6 prefix and prefix length in the RADIUS packet. Use a matching prefix length for the sticky lookup of end user packets. Enter a prefix length from 1 to 128. The default is 64.
Wait For Acknowledgement[1]	Check this check box to configure the ACE to reload-balance RADIUS requests that hit framed-ip sticky entries (excluding the real server in sticky entry) when the Accounting-Start does not receive a response. This feature is designed for scenarios in which sticky entries are created during the Accounting phase. Clear this check box to configure the ACE not to use the wait for an acknowledgement feature.

1. This field requires ACE software Version A5(2.1) or later.

RTSP Header Sticky Group Attributes

Table 9-9 describes the RTSP header sticky group attributes.

Table 9-9 RTSP Header Sticky Group Attributes

Field	Description
Offset	Number of bytes that the virtual server is to ignore starting with the first byte of the cookie. Valid entries are from 0 to 999. The default is 0 (zero), which indicates that the virtual server does not exclude any portion of the cookie.
Length (Bytes)	Length of the portion of the cookie (starting with the byte after the offset value) that the ACE is to use for sticking the client to the server. Valid entries are from 1 to 1000. The default is 1000.

SSL Header Sticky Group Attributes

Note This SSL sticky option requires ACE software Version A5(2.0) or later.

Table 9-10 describes the SSL header sticky group attributes.

Table 9-10 SSL Sticky Group Attributes

Field	Description
Enable Sticky For Response	Check the checkbox to instruct the ACE to parse the response bytes from a server and perform sticky learning. Clear the checkbox when you do not want the ACE to perform this operation.
Length (Bytes)	Length of the SSL session ID that needs to be parsed. Valid entries are integers from 1 to 1000.

Displaying All Sticky Groups by Context

You can display all sticky groups associated with a virtual context.

Procedure

Step 1 Choose **Config > Devices**.

The Virtual Contexts table appears.

Step 2 In the Virtual Contexts table, choose the virtual context with the sticky groups that you want to display, and choose **Load Balancing > Stickiness**.

The Sticky Groups table appears, listing the sticky groups associated with the selected context.

Step 3 Do the following:

- Choose a sticky group and click the **Show Sticky Database...** button to view a popup window that displays the output of the **show sticky database group** <*name*> **detail** command.
- Choose a sticky group whose client sticky entries you want to delete and click the **Clear Sticky Database** button.

  A message appears asking you to confirm the clearing of the sticky entries.

  Clearing the sticky entries impacts all virtual servers associated with the selected sticky database group.

Note The **Show Sticky Database...** button is displayed only for ACE software version A5(1.0) or later.

Related Topics

- Configuring Sticky Groups, page 9-12
- Configuring Sticky Statics, page 9-23

Configuring Sticky Statics

You can configure sticky statics.

Note Fields and information related to IPv6 require ACE module and ACE appliance software Version A5(1.0) or later.

Assumption

A sticky group has been configured. See the "Configuring Sticky Groups" section on page 9-12 for more information.

Procedure

Step 1 Choose **Config > Devices >** *context* **> Load Balancing > Stickiness**.

The Sticky Groups table and Sticky Statics tab appears. If you do not see the Sticky Statics tab beneath the Sticky Groups table, click the **Switch between Configure and Browse Modes** button.

Step 2 From the Sticky Groups table, choose the sticky group that you want to configure for sticky statics

Step 3 From the Sticky Statics tab, click **Add** to add a new entry to the table, or select an existing entry, then click **Edit** to modify it.

The Sticky Statics configuration screen appears.

Step 4 In the Sequence Number field, either accept the automatically incremented number for this entry or enter a new sequence number.The sequence number indicates the order in which multiple sticky static configurations are applied.

The sequence number indicates the order in which multiple sticky static configurations are applied.

Step 5 From the Type drop-down list, choose the sticky group type.

The choices are as follows:

- **HTTP Content**—The ACE sticks client connections to the same real server based on a string in the data portion of the HTTP packet.
- **HTTP Cookie**—The ACE either learns a cookie from the HTTP header of a client request or inserts a cookie in the Set-Cookie header of the response from the server to the client, and then uses the learned cookie to provide stickiness between the client and server for the duration of the transaction.
- **HTTP Header**—The ACE sticks client connections to the same real server based on HTTP headers.

- **IP Netmask**—The ACE sticks a client to the same server for multiple subsequent connections as needed to complete a transaction using the client source IP address, the destination IP address, or both based on the IPv4 netmask. You can optionally configure an IPv6 prefix length with this sticky type.

  Note If an organization uses a megaproxy to load balance client requests across multiple proxy servers when a client connects to the Internet, the source IP address is no longer a reliable indicator of the true source of the request. In this situation, you can use cookies or another sticky method to ensure session persistence.

- **V6 Prefix**—(Option that appears only for ACE module and ACE appliance software Version A5(1.0) or later) The ACE sticks a client to the same server for multiple subsequent connections as needed to complete a transaction using the client source IP address, the destination IP address, or both based on the IPv6 prefix length. You can optionally configure an IPv4 netmask with this sticky type.
- **Layer 4 Payload**—The ACE sticks client connections to the same real server based on a string in the payload portion of the Layer 4 protocol packet.
- **RADIUS**—The ACE sticks client connections to the same real server based on a RADIUS attribute.
- **RTSP Header**—The ACE sticks client connections to the same real server based on the RTSP Session header field.
- **SIP Header**—The ACE sticks client connections to the same real server based on the SIP Call-ID header field.

Step 6 If you chose HTTP Cookie, HTTP, RTSP, or SIP Header for the sticky type, in the Static Value field, enter the cookie string value.

Valid entries are unquoted text strings with a maximum of 255 alphanumeric characters. If the string includes spaces, enclose the string with quotes.

Step 7 If you chose IP Netmask or V6 Prefix for the sticky type, do the following:

a. For the IP Address Type, select either IPv4 or IPv6.

b. In the Static Source field, enter the source IP address of the client.

c. In the Static Destination field, enter the destination IP address of the client.

Step 8 In the Named Real Server field, choose the real server to associate with this static sticky entry.

Step 9 In the Port field, enter the port number of the real server.

Valid entries are from 1 to 65535.

Step 10 Do one of the following:

- Click **Deploy Now** to deploy this configuration on the ACE and save your entries to the running-configuration and startup-configuration files.
- Click **Cancel** to exit the procedure without saving your entries and to return to the Sticky Statics table.
- Click **Next** to deploy your entries and to configure another sticky static entry.

Related Topics

Configuring Sticky Groups, page 9-12