Configuring WAAS with Akamai Connect

This chapter describes how to configure Cisco WAAS with Akamai Connect, which is an integrated solution that combines WAN optimization and intelligent object caching to accelerate HTTP/S applications, video, and content.

Note Throughout this chapter, the term Cisco WAAS device is used to refer collectively to the Cisco Wide Area Application Services (Cisco WAAS) Central Managers and Cisco Wide Area Application Engines (WAEs) in your network. The term WAE refers to WAE and Cisco Wide Area Virtualization Engine (WAVE) appliances, and Cisco Virtual WAAS (vWAAS) instances.

This chapter contains the following sections:

Note Akamai Connect is the HTTP/S object cache component added to Cisco WAAS, integrated into the existing WAAS software stack and leveraged via the HTTP Application Optimizer. WAAS with Akamai Connect helps to reduce latency for HTTP/S traffic for business and web applications.Akamai Connected Cache is a component of Akamai Connect, which allows the Cache Engine to cache content that is delivered by an Edge server on the Akamai Intelligent Platform.

This chapter contains the following sections:

About Cisco WAAS with Akamai Connect

Akamai Connect is the HTTP/S object cache component added to Cisco WAAS, integrated into the existing WAAS software stack and leveraged via the HTTP Application Optimizer.

- Cisco WAAS with Akamai Connect helps to reduce latency for HTTP/S traffic for business and web applications, and can improve performance for many applications, including Point of Sale (POS), HD video, digital signage, and in-store order processing.
- Cisco WAAS with Akamai Connect provides significant and measurable WAN data offload, and is compatible with existing WAAS functions such as DRE (deduplication), LZ (compression), TFO (Transport Flow Optimization), and SSL acceleration (secure/encrypted) for first and second pass acceleration.

Akamai Connected Cache is a component of Akamai Connect, which allows the cache engine to cache content that is delivered by an edge server on the Akamai Intelligent Platform.

The following list highlights some of the benefits offered by Cisco WAAS with Akamai Connect:

- Intelligent transparent object caching.
- Seamless integration of Akamai Connect into Cisco WAAS software and configuration, using either the Cisco WAAS Central Manager or Cisco WAAS CLI.
- Integration with Akamai's Edge Grid Network, which provides low-latency Content Delivery Network transfers via Akamai Connected Cache.
- Significant and measurable WAN data offload.
- Cache prepositioning (warming) for websites that you specify.
- Hostname rules for cache control of specific websites or domains.
- First-pass and second-pass acceleration, with Akamai Connect working with Cisco WAAS middle-mile capabilities, including DRE, LZ, TFO, and SSL acceleration.
- Dual-sided or single-sided deployment.

Components of Cisco WAAS with Akamai Connect

The table provides overviews of Cisco WAAS with Akamai Connect components, links to further information, and links to Akamai Connect configuration procedures.

Table 1: Components of Cisco WAAS with Akamai Connect

Component	Description and Further Information
Deployment Options	You can deploy Cisco WAAS with Akamai Connect as a dual-sided or single-sided deployment. For more information, see Deployment Options for Cisco WAAS with Akamai Connect, on page 4.
Akamai Connect license	The Akamai Connect license for Cisco WAAS is an advanced license available for all supported Cisco with Akamai Connect devices. The Akamai Connect license for Cisco WAAS is aligned with the number of optimized connections in each supported Cisco WAAS device. For more information, see Activating and Managing the Akamai Connect License, on page 14.
Supported Cisco WAAS platforms	For Cisco WAAS Version 5.4.1 and later, Cisco with Akamai Connect supports WAAS and vWAAS devices up to 6,000 connections. For Cisco WAAS Versions later than Cisco WAAS Version 5.4.1, Cisco with Akamai Connect supports WAAS and vWAAS devices beyond 6,000 connections. For more information, see Supported Platforms for Cisco WAAS with Akamai Connect, on page 5.
Transparent cache and caching policies	The Transparent cache, Akamai's high-performance HTTP object cache, provides the ability to locally cache HTTP-based content for LAN-like performance, regardless of whether the web application was served from the private corporate cloud or the public Internet. This content includes on-demand and live HTTP video streams to deliver fast, high-quality, high-definition video experiences in the branch, all while offloading the enterprise network. There are four caching policies (modes): **Basic**, **Standard** (default), **Advanced**, and **Bypass**. For more information, see Setting Transparent Caching Policies, on page 23.
Akamai Connected Cache	Akamai's proprietary caching rules in connection with the edge servers of the Akamai Intelligent Platform lets you cache and deliver content inside the branch office that might otherwise be deemed noncacheable. This content could be an enterprise's own web content or any content that is delivered by the Akamai Intelligent Platform, which is up to thirty percent of all web traffic. For more information, see Enabling Akamai Connected Cache.
Over the Top (OTT) caching	Over-The-Top (OTT) caching is used for streamed content, particularly video content. OTT caching caches HTTP content served from dynamic URLs and content marked as noncacheable, such as YouTube videos. Akamai achieves this by using metadata logic to determine a unique cache key per video, which allows dynamic URLs to be cached. For more information, see Enabling Over the Top (OTT) Caching, on page 21.

Component	Description and Further Information
Cisco Cloud Web Security (CWS)	Cisco Cloud Web Security (CWS) provides content scanning of HTTP and HTTP/S traffic, and provides malware protection service to web traffic. CWS enforces content filtering by enabling force IMS for every cached object, for both single-sided and dual-sided deployment. For more information, see Enabling Cisco Cloud Web Security (Cisco CWS), on page 27.
Cisco WAAS connections to the Akamai network	There are three ways for Cisco WAAS devices to connect to the Akamai network: • No HTTP proxy • Use Cisco WAAS Central Manager as HTTP proxy • Use an external HTTP proxy For more information, see Configuring Cisco WAAS Connections to the Akamai Network, on page 28.
Cache prepositioning	Cache prepositioning, also known as cache warming, allows you to specify a policy to prefetch and cache content at a specified time. Cache prepositioning allows you to take advantage of idle time on the WAN to transfer large or frequently accessed files to selected Cisco WAAS devices, so that users can benefit from cache-level performance even during first-time access of these files. For more information, see Configuring Akamai Connect Cache Prepositioning, on page 35.
Cisco support for Microsoft Update	Cisco support for Microsoft Windows Update enables the Akamai cache engine to support Windows Update in two ways: to download and cache full objects even when ranges within objects that not in cache are requested, and future range requests on the objects can be served out of cache. For more information, see Cisco Support for Microsoft Windows Update, on page 42.

Deployment Options for Cisco WAAS with Akamai Connect

This section contains the following topics:

About Deployment Options for Cisco WAAS with Akamai Connect

You can deploy Cisco WAAS with Akamai Connect as a dual-sided or single-sided deployment:

- Dual-sided deployment of Cisco WAAS with Akamai Connect provides the following benefits for HTTP and HTTPS traffic:
  - Transparent caching of customer-owned, Intranet web resources.

- Caching in branch only.
- Includes prepositioning (for non-SSL content).

- Single-sided deployment of Cisco WAAS with Akamai Connect provides the following benefits for HTTP and HTTPS traffic:
  - Generic web resources that utilize proxy-specific HTTP cache-control headers.
  - Caching in branch only.
  - Includes prepositioning (for non-SSL content).

Note For Transparent caching in **Standard** mode, single-sided deployment of Cisco WAAS with Akamai Connect is enabled by default.

Operating Guidelines for Cisco WAAS with Akamai Connect

Consider the following operating guidelines for Cisco WAAS with Akamai Connect:

- There is no separate cache for HTTPS content. However, data is stored differently for the same site if both HTTP and HTTPS are accessing. (The way the sites are stored in the cache is based on the URL, and this will change between HTTP and HTTPS.)
- You cannot view the contents of the cache, and cannot pin content to make it remain in the cache, for example, for prepositioned content.
- The Akamai cache engine has no explicit integration with Cisco AppNav. The Cisco AppNav status is based on the Cisco HTTP application accelerator.

Supported Platforms for Cisco WAAS with Akamai Connect

The flow of allocated resources to the Akamai Cache Engine is controlled by the WAAS Central Manager, but the overall resource pool and the amount of resources that can be allocated to the Akamai Cache Engine is controlled by the hardware platform, and the number of supported connections and users that the router is designed to service.

This section contains the following topics:

Supported Platforms for Cisco WAAS with Akamai Connect up to 6,000 Connections

The flow of allocated resources to the Akamai cache engine is controlled by the Cisco WAAS Central Manager, but the overall resource pool and the amount of resources that can be allocated to the Akamai cache engine is controlled by the following:

- the hardware platform

- the number of supported connections and users that the router is designed to service

The following table shows supported platforms for Cisco WAAS with Akamai Connect up to 6,000 connections, for Cisco WAAS Version 5.4.1 and later. For information on supported platforms for Cisco WAAS with Akamai Connect beyond 6,000 connections, see Supported Platforms for Cisco WAAS with Akamai Connect beyond 6,000 Connections.

Table 2: Supported Platforms for Cisco WAAS with Akamai Caching up to 6,000 Connections

Appliance	SM	vWAAS	ISR-WAAS
N/A	N/A	vWAAS-150	ISR-G2 and ISR-G3
WAVE-294	SM-700	vWAAS-200	ISR-WAAS-750 • ISR-4451 • ISR-4431 • ISR-4351 • ISR-4331 • ISR-4321
WAVE-594	SM-900	vWAAS-750	ISR-WAAS-1300 • ISR-4451 • ISR-4431
WAVE-694	SM-710	vWAAS-1300	ISR-WAAS-2500 • ISR-4451
N/A	SM-910	vWAAS-2500	N/A
N/A	N/A	vWAAS-6000	N/A

Note If you are upgrading from a version earlier than Cisco vWAAS in Cisco WAAS Version 5.4.x, you will need a third disk and possibly more memory added. For more information, see the "Cisco vWAAS with Akamai Connect" chapter in the *Cisco Virtual Wide Area Application Services Configuration Guide*.

Supported Platforms for Cisco WAAS with Akamai Connect beyond 6,000 Connections

This section describes the supported Cisco platforms for Cisco WAAS with Akamai Connect beyond 6,000 Connections, and operating guidelines for these platforms.

The flow of allocated resources to the Akamai cache engine is controlled by the Cisco WAAS Central Manager, but the overall resource pool and the amount of resources that can be allocated to the Akamai cache engine is controlled by the hardware platform and the number of supported connections and users that the router is designed to service.

For Cisco WAAS Version 6.2.1 and later, the following list shows the Cisco WAAS with Akamai Connect supported platforms for scaling beyond 6,000 connections:

- Cisco WAVE-7541 or Cisco CSP 5228-W
- Cisco WAVE-7571 or Cisco CSP 5228-W
- Cisco WAVE-8541 or Cisco CSP 5436-W
- Cisco vWAAS-12000
- Cisco vWAAS-50000

For Cisco WAAS with Akamai Connect in Cisco WAAS Version 5.4.1 and earlier, see Supported Platforms for Cisco WAAS with Akamai Connect up to 6,000 Connections.

Consider the following operating guidelines for supported platforms for Cisco WAAS with Akamai Connect beyond 6,000 connections:

- Supported Cisco WAVE models with Akamai Connect beyond 6,000 Connections:
  - The table "Cisco WAAS with Akamai Connect Requirements for HTTP Object Cache" shows the supported Cisco WAVE models with Akamai Connect beyond 6,000 connections, and specifications for the total HTTP object cache connections and cache engine cache disk.

Table 3: Cisco WAAS with Akamai Connect Requirements for HTTP Object Cache

Cisco WAVE Model	Total HTTP Object Cache Connections	Cache Engine Cache Disk
WAVE-7541	18 K	708 GB
WAVE-7571	45 K	839 GB
WAVE-8541	112 K	675 GB

  - The Akamai cache engine connection-handling capacity is determined by the upper limit of memory that is given to the Akamai cache engine at startup. The Akamai cache engine will allocate memory as needed up to the upper limit. In case of overload, the connection will be optimized by HTTP-AO, without a caching benefit.

Note When a Cisco WAVE model used for Akamai Connect beyond 6,000 connections is assigned to a device group in the Cisco WAAS Central Manager *after* Akamai Connect is already enabled, you must manually reload the Cisco WAVE device. Akamai Connect will remain in shutdown state until the reload is performed.

- Supported Cisco vWAAS models with Akamai Connect beyond 6,000 connections:

- The table "Cisco vWAAS with Akamai Connect Requirements for Beyond 6,000 Connections" shows supported Cisco vWAAS models with Akamai Connect beyond 6,000 connections, and specifications for total HTTP object cache connections, cache engine cache disk, and additional resources that may be needed.

Table 4: Cisco vWAAS with Akamai Connect Requirements for Beyond 6,000 Connections

Cisco vWAAS Model	Total HTTP Object Cache Connections	Cache Engine Cache Disk	Additional Resource to be Added
vWAAS-12000	12 K	750 GB	6 GB RAM, 750 GB disk
vWAAS-50000	50 K	850 GB	850 GB disk

- For Cisco vWAAS-12000 and Cisco vWAAS-50000:
  - HTTP object cache will scale up to the platform TFO limit. To achieve this, you must augment the platform resources (CPU, RAM, and disk) during provisioning.
  - For Cisco vWAAS-12000 and Cisco vWAAS-50000, you must allocate Akamai cache engine cache disk resources. Cache disk requirements are shown in above table "Cisco vWAAS with Akamai Connect Requirements for Beyond 6,000 Connections."
  - For Cisco vWAAS-12000, you must allocate at least 6 GB of additional RAM.

- Consider the following overview of operating guidelines for Cisco vWAAS with Akamai Connect caching. For detailed information on configuring and using Cisco vWAAS with Akamai Connect caching, see the “Cisco vWAAS with Akamai Connect” chapter in the *Cisco Virtual Wide Area Application Services Configuration Guide*.
  - For Cisco vWAAS in Cisco WAAS Version 6.1.1 and later, Cisco vWAAS-150 on Cisco ISR-WAAS is supported for Akamai Connect.
  - For Cisco vWAAS in Cisco WAAS Version 6.2.1 and later, Cisco vWAAS-150 is also supported for RHEL KVM and Microsoft Hyper-V.
  - For vWAAS in Cisco WAAS versions earlier than 6.x, Akamai Connect beyond 6,000 connections is not supported for Cisco vWAAS on RHEL KVM or KVM on CentOS.

Configuring HTTP Object Cache

This section contains the following topics:

Configuring HTTP Object Cache in Cisco Devices with Akamai Connect

Before you begin

- HTTP object cache is used to enable the Akamai cache engine for a Cisco device, for Cisco WAAS Version 6.2.1 and later.

- Enabling HTTP object cache for Cisco WAVE-7541, Cisco WAVE-7571 or Cisco WAVE-8541 includes configuring the device profile feature, repartitioning the Cisco WAVE data disk, and if needed, upgrading your Cisco WAAS system to Cisco WAAS Version 6.2.1 or later.

Procedure

Step 1 If needed, upgrade the Cisco WAAS Central Manager and Cisco WAE devices to Cisco WAAS Version 6.2.1 or later.

- For complete upgrade instructions, including critical prerequisites before upgrading to Cisco WAAS Version 6.2.1 or later, see the Release Note for Cisco Wide Area Application Services for your Cisco WAAS version.
- To configure HTTP object cache on Cisco vWAAS-12000 or Cisco vWAAS-50000, and to avoid object and DRE caching being lost due to execution of the **disk delete-data-partitions** EXEC command, you must downgrade from WAAS Version 6.2.x to WAAS Version 5.x, and then upgrade to WAAS Version 6.2.x.

Step 2 To enable HTTP object cache on the Cisco WAVE device, run the **accelerator http object-cache enable** global configuration command.

A message is displayed to restart the system, with two prerequisite procedures:

a) Run the **disk delete-data-partitions** EXEC command.
b) Enable **Device Profile**.

You must provide approval for each of these procedures.

Step 3 Run the **disk delete-data-partitions** EXEC command.

To accommodate the larger-scale connections available for Cisco WAAS Version 6.2.1 and later with Akamai Connect, the single partition for the RAID5-based disk subsystem is split into multiple partitions.

Note The **disk delete-data-partitions** EXEC command deletes all data partitions on all logical drives, including CONTENT, PRINTSPOOL, and SYSFS partitions. These partitions include all DRE and SMB object cache files, SYSFS and print spool files. New partitions are created at system restart.

Step 4 Enable **Device Profile**: After the upgrade is complete, **Device Profile** is initially disabled.

Considering the following operating guidelines for **Device Profile**:

- For Cisco WAAS Version 6.2.1 and later, **Device Profile** enables the device mode as branch, which tunes the resource allocation for various Cisco WAAS services as a branch traffic scenario and branch services.
- For Cisco WAVE-7541 and Cisco WAVE-8541, the **Device Profile** is automatically set or unset when you enable or disable HTTP object cache.
- For Cisco WAVE-7571, the **Device Profile** feature requires you to reboot the system to change the Device Profile feature status.

You can enable Device Profile from the Cisco WAAS Central Manager or the Cisco WAAS CLI.

- To enable Device Profile from the Cisco WAAS Central Manager:

a. Choose **Device** > *device-name* > **Configure** > **Caching** > **Device Profile**.

The **Device Profile** window is displayed.

b. To enable **Device Profile**, check the **Branch** check box.

Note The **Device Profile** feature is enabled at the individual device level; it is not enabled for an entire device group.

c. Click **Submit**.

- To enable **Device Profile** from the Cisco WAAS CLI:
  - To configure the device to function as a branch device, and to configure resource pre-allocation resources for various WAAS services to be branch traffic scenario and branch services, run the **device mode application-accelerator profile branch** global configuration command.

Step 5 Restart the system.

When you restart the system using the Cisco WAAS Central Manager, the HTTP object cache is enabled on the device.

Step 6 Enable Akamai Connect.

For WAVE models 7541 and 8541, the **Device Profile** feature is automatically set/unset when you enable/disable HTTP OC. For WAVE-7571, the **Device Profile** setting requires you to reboot to change the **Device Profile** feature status.

Downgrading and Upgrading a Cisco vWAAS Device with Additional Akamai Cache Disk Removed and Reinstalled

Procedure

Step 1 For the Cisco vWAAS device in Cisco WAAS Version 6.2.x with Akamai Connect enabled:

a) From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, and then choose **Configure** > **Caching** > **Akamai Connect**.
b) To disable Akamai Connect, uncheck the **Enable Akamai Connect** check box.
c) Power down the Cisco vWAAS device.

Step 2 Remove the additional Akamai Cache disk.

Step 3 Power on the Cisco vWAAS device.

Step 4 Downgrade from Cisco WAAS Version 6.2.x to Cisco WAAS Version 5.x.

Step 5 Upgrade the Cisco WAAS Central Manager and Cisco WAE devices to Cisco WAAS Version 6.2.x.

Step 6 After the upgrade is complete, power off the device.

Step 7 Reinstall the additional Akamai Cache disk.

Step 8 Power on the Cisco vWAAS device.

Step 9 Enable Akamai Connect.

a) From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure > Caching > Akamai Connect**.
b) To enable Akamai Connect, check the **Enable Akamai Connect** check box.
c) Click **Submit**.

Step 10 Enable HTTP object cache from the Cisco WAAS Central Manager or from the Cisco WAAS CLI.

Consider the following guidelines for enabling HTTP object cache:

- A message is displayed regarding the required additional memory and disk resources. For resource guidelines, see the "Cisco WAAS with Akamai Connect Requirements for HTTP Object Cache" table.
- For information on enabling HTTP object cache for Cisco WAVE devices, see Configuring HTTP Object Cache.
- For information on enabling HTTP object cache for Cisco vWAAS devices, see Configuring HTTP Object Cache in Cisco Devices with Akamai Connect.

Step 11 Power down the Cisco vWAAS device and add the necessary resources to the Cisco vWAAS device.

Step 12 Power up the Cisco vWAAS VM.

HTTP object cache is enabled on the Cisco vWAAS device.

Downgrading and Upgrading a Cisco vWAAS Device with Additional Akamai Cache Disk Remaining In Place

Procedure

Step 1 Upgrade the Cisco WAAS Central Manager and Cisco WAE devices to Cisco WAAS Version 6.2.1 or later.

Step 2 Disable Akamai Connect.

a) From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, and then choose **Configure > Caching > Akamai Connect**.

The **Akamai Connect** window appears, with the **Cache Settings** tab displayed.

b) To disable Akamai Connect, uncheck the **Enable Akamai Connect** check box.
c) Power down the Cisco vWAAS device.

Step 3 Downgrade from Cisco WAAS Version 6.2.x to Cisco WAAS Version 5.x.

Step 4 Upgrade the Cisco WAAS Central Manager and Cisco WAE devices to Cisco WAAS Version 6.2.x.

Step 5 Run the **disk delete-data-partitions** EXEC command and restart the system.

a) From the Cisco WAAS CLI, a message is displayed to run the **disk delete-data-partitions** EXEC command and restart the system.

The Cisco WAAS Central Manager does not display this message.

b) After the upgrade, you must run the disk delete-data-partitions command to enable Akamai Connect.

Note The **disk delete-data-partitions** EXEC command deletes all data partitions on all logical drives, including CONTENT, PRINTSPOOL, and SYSFS partitions. These partitions include all DRE and SMB object cache files, SYSFS and print spool files. New partitions are created at system restart.

Step 6 Enable Akamai Connect.

a) From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure** > **Caching** > **Akamai Connect**.

The Akamai Connect window appears, with the **Cache Settings** tab displayed ().

b) To enable Akamai Connect, check the **Enable Akamai Connect** check box.

c) Click **Submit**.

Workflow for Enabling and Using Cisco WAAS with Akamai Connect

The following table shows the workflow for enabling and using Cisco WAAS with Akamai Connect.

Table 5: Workflow for Enabling and Using Cisco WAAS with Akamai Connect

Task	Description and Links to Associated Tasks
1. Receive and activate the Akamai Connect license.	• Activating and Managing the Akamai Connect License, on page 14 • About the Akamai Connect License, on page 14 • Prerequisites for Activating the Akamai License, on page 14 • Activating the Akamai Connect License File, on page 15
2. Enable Akamai Connect.	• Enabling Akamai Connect, on page 18 • Confirming Akamai Connect Configuration Prerequisites, on page 18 • Turning on the Akamai Cache Engine and Enabling Akamai Connect, on page 19
3. Enable Akamai Connected Cache.	• Enabling Akamai Connected Cache, on page 20 • About Akamai Connected Cache, on page 20 • Procedure for Enabling Akamai Connected Cache, on page 20

Task	Description and Links to Associated Tasks
4. Enable Over the Top (OTT) caching.	• Enabling Over the Top (OTT) Caching, on page 21 • About OTT Caching, on page 21 • Procedure for Enabling OTT Caching, on page 22
5. Set transparent caching policies.	• Setting Transparent Caching Policies, on page 23 • Transparent Caching and Caching Modes, on page 23 • Setting a Transparent Caching Policy for All Sites, on page 25 • Setting a Transparent Caching Policy for a Specific Site, on page 26
6. Enable Cisco Cloud Web Security (CWS).	• Enabling Cisco Cloud Web Security (Cisco CWS), on page 27 • About Cisco CWS, on page 27 • Procedure for Enabling Cisco CWS, on page 28
7. Configure Cisco WAAS connections to the Akamai network.	• Configuring Cisco WAAS Connections to the Akamai Network, on page 28 • About Cisco WAAS Connections to the Akamai Network, on page 28 • Configuring No HTTP Proxy, on page 29 • Configuring the Cisco WAAS Central Manager as HTTP Proxy, on page 30 • Configuring External HTTP Proxy, on page 31
8. Configure Server Address Validation.	• Configuring Server Address Validation, on page 32 • About Server Address Validation, on page 32 • Alarms Used with Server Address Validation, on page 33 • Configuring Server Address Validation, on page 32
9. Configure Akamai Connect cache prepositioning.	• Configuring Akamai Connect Cache Prepositioning, on page 35 • Configuring a Cache Prepositioning Task, on page 35 • Viewing Cache Prepositioning Task Status, on page 38 • Copying Cache Prepositioning Tasks, on page 39

Task	Description and Links to Associated Tasks
10. Configure Akamai Connect HTTP/S preposition proxy.	• Configuring HTTP/S Preposition Proxy for Akamai Connect, on page 40 • About HTTP/S Preposition Proxy for Akamai Connect, on page 40 • Configuring Global Proxy Host and Port for Preposition Tasks, on page 40 • Modifying Proxy Settings for an Individual Prepositioning Task, on page 41 • Removing Proxy Settings for an Individual Prepositioning Task, on page 41

Activating and Managing the Akamai Connect License

This section contains the following topics:

About the Akamai Connect License

The Akamai Connect license for Cisco WAAS is an advanced license available for all supported Cisco with Akamai Connect devices. The Akamai Connect license for Cisco WAAS is aligned with the number of optimized connections in each supported Cisco WAAS device.

Prerequisites for Activating the Akamai License

Before you upload a new Akamai license or before you enable Akamai Connect on Cisco WAAS and activate the Akamai Connect License file, consider these guidelines:

1. Before you upload a new Akamai license, collect or confirm the following Akamai License Customer ID information:
   - The new license activation file to be uploaded.
   - A customer ID snapshot from the Cisco WAAS Central Manager Akamai Diagnostics: Choose **Home > Monitor > Troubleshoot > Akamai Diagnostics > Akamai Connect License > Details**.
   - To capture the hostname and Akamai ID, copy the list of some devices, as either a snapshot or an Excel spreadsheet: Choose **Home > Monitor > Troubleshoot > Akamai Diagnostics > Akamai Connect License > Details > Test**.
   - Open a service request with this information so that Cisco TAC can assist you further. For further information on contacting TAC, see the Cisco Support and Downloads page, Contacts/Support Cases section.
2. Before you enable Akamai Connect on Cisco WAAS and activate the Akamai Connect License file, complete the following prerequisites:

- Confirm the readiness of your Cisco WAAS configuration, as described in the first bulleted text of this section.
- For information on the status of an active Akamai Connect license, see Akamai Connect Diagnostics Using the Central Manager in the chapter "Troubleshooting Your Cisco WAAS Network."

Activating the Akamai Connect License File

Procedure

Step 1 Enable Akamai Connect, as described in Workflow for Enabling and Using Cisco WAAS with Akamai Connect.

If this is the first time you are enabling Akamai Connect, you are prompted to provide the activation file for licensing.

Step 2 If you have not yet done so, purchase an Akamai Connect license from your Cisco account representative or reseller. The following actions are generated by this purchase:

- The account representative or reseller enters the order into the Cisco Commerce Workspace (CCW) system. The order *must* specify an email address for eDelivery of the Activation file.
- CCW contacts the Akamai Luna Portal to request a license or licenses for the number and type of Akamai licenses entered.
- Akamai generates and sends the license(s) to the CCW system in the form of a single activation file.
- The CCW system sends an email, with the activation file attached, to the email address specified in the order. The order of priority for selecting the email address in a CCW order is:
  - **Priority1**: eDelivery email address
  - **Priority2**: end customer email address
  - **Priority3**: shipping contact email address

  Note If you do not provide an email address in your order, you will not receive an activation file.

Step 3 To upload the **Akamai Connect License file**, choose **Home > Admin > Licenses > Akamai Connect**.

The **Upload Akamai Connect License** file window is displayed.

Step 4 Use to the **Browse** button to highlight and select the activation file, and click **Upload**.

- The authentication data in the activation file is transmitted to the Akamai Luna portal.
- After the device message is sent to the Akamai Luna portal, the Akamai Luna portal sends the Entitlement Code to the Cisco WAAS Central Manager and the Akamai Management Gateway (AMG). The Cisco WAAS Central Manager sends the Entitlement Code to Cisco WAAS, and the AMG rolls out the Entitlement Code to the edge servers on the Akamai Grid Network.

  Each of these steps happens automatically, but each takes some time to complete.

- The Entitlement Code is maintained on the Akamai Luna portal, on the AMG, and on the Cisco WAAS device. Cisco WAAS connects to the AMG using a proxy/DNS server that can resolve the address **amg.terra.akamai.com**.

Step 5 The activation process begins.

The **Status of Devices with Akamai Connect Feature Configured** table listing displays the following types of status for one, some, or all devices. The Table 6: Status Indicator States for Device, Operational, and Connectivity Status table shows the states through which the indicators proceed: **Akamai Device Status**, **Operational Status**, and **Connectivity to Akamai**.

Table 6: Status Indicator States for Device, Operational, and Connectivity Status

Status Category	First Status	Second Status	Third Status	Fourth Status
Akamai Device Status	ActivationInProgress	ActivationInProgress	Active	Active
Operational Status	Disconnected	Connected	Connected	Connected
Connectivity to Akamai	Activating	Activating	Activated	Connected

Note The activation process for WAAS devices may take between 15-60 minutes to complete, and for this time period, the **Connectivity to Akamai** status displays as **Activating**. During this time, device(s) may not be able to communicate with the Akamai Network, because they are not recognized by the AMG until the activation process is complete, and the **Connectivity to Akamai** status displays as **Connected**.

Step 6 For the final steps in the registration process:

- Akamai Luna sends the Akamai Connected Cache credentials to the AMG and to the Akamai edge servers on the Akamai Grid network. The AMG forwards the Akamai Connected Cache credentials on to Cisco WAAS.
- With the Akamai Connected Cache credentials on both Cisco WAAS and the Akamai edge servers, the Akamai Connected Cache is enabled, and caching requests can be served by the Akamai edge servers. This authenticated connection can then service requests for Akamai Connected Cache and OTT caching from the Akamai Grid network Akamai edge servers.
- The registration of each Cisco WAE begins. The Cisco WAAS Central Manager provides information to the Akamai Luna Portal for each Cisco WAAS device that will be running Akamai Connect.

  Note The **Connected** Operational Status can take several minutes to complete. Rollout of the activation to the Akamai edge servers can take up to 45 minutes to complete. A device may take from a few minutes to up to two hours to show an **Active** Activation Status, depending on when the request was made, traffic conditions, and other variables.

Step 7 Each Cisco WAE that has been sent the entitlement code will try to make an SSL connection to the AMG using **amg.terra.akamai.com**. The Akamai Luna Portal will push out the Akamai Connected Cache credentials to the AMG and to the Akamai Grid Network (to the Akamai edge servers).

- The AMG will push the Akamai Connected Cache credentials out to each of the Cisco WAEs that are configured for Akamai Connected Cache. If OTT is enabled, the OTT metadata needed to help cache YouTube objects is also processed at this time.
- The Akamai Connected Cache credentials are sent by the Cisco WAE cache engine when going to the origin server. If the Cisco WAE cache engine has valid credentials according to the Akamai edge server, the Akamai edge server then provides objects to the Cisco WAE cache engine that are not normally cacheable to other devices.

Step 8 The Cisco WAE cache engine will request new credentials daily and will be good for two days. The connections are always established from the Cisco WAE or Cisco WAAS Central Manager over TCP 443 to the AMG.

- For security, firewalls are usually deployed by performing statefull inspection on traffic from within the company to the outside. They are also configured to block unknown traffic from the outside to the inside.

  Because connection should not initiate from AMG to any Cisco WAAS Central Manager or Cisco WAE at any time, there should not be an issue. If there is, then a hole will need to be made to allow the Cisco WAAS Central Manager or Cisco WAE to communicate with any device on port 443.
- The Devices listing on the All Devices window includes a column titled Akamai Connect, which shows the status of each device: **Active**, **Not Supported**, **Connected**, or **Disconnected**.

Step 9 As needed, configure HTTP proxy or external HTTP proxy, described in Configuring Cisco WAAS Connections to the Akamai Network.

Deregistering and Reregistering a Cisco WAAS Device

This section provides an overview of how to deregister and reregister a Cisco WAAS device. For more information, see Changing Device Mode in the chapter "Planning Your Cisco WAAS Network."

- To change the device mode of a Cisco WAAS device that is already registered with a Cisco WAAS Central Manager, you must perform the following tasks:
  1. Deregister the Cisco WAAS device from the Cisco WAAS Central Manager.
  2. Change the device mode of the Cisco WAAS device.
  3. Reload the Cisco WAAS device.
  4. Re-enable CMS services for the Cisco WAAS device.
- When you deregister a Cisco WAAS device from the Cisco WAAS Central Manager:
  - The Cisco WAAS Central Manager triggers the removal of the device record on the Akamai side, thereby invalidating the entitlement key used by the Cisco WAE cache engine to talk to AMG devices.
  - On the Cisco WAAS side, the Cisco WAE cache engine will continue to operate in **Transparent** caching mode.
- When you reregister a Cisco WAAS device with the Cisco WAAS Central Manager, one of two things happen:

- The Cisco WAAS Central Manager auto-assigns the Cisco WAAS device to device groups (that are so marked). If any of these device groups have Akamai Connect and HTTP cache settings, the Cisco WAAS Central Manager will trigger registration with Akamai.
- If no device group is configured with Akamai Connect and HTTP cache settings, the registration is done individually.
- After the Cisco WAAS device is registered, it will get a new entitlement key.

Replacing an Inactive or Expired Akamai Connect License

Procedure

Step 1 When a license is inactive or expired, a notification is displayed in one of these Cisco WAAS Central Manager windows:

- The following notification is displayed in the **Home > Admin > Akamai Connect** window: **Akamai Connect License is Inactive. Please remove current license and import valid license.**
- The following notification is displayed in the **Home > Monitor > Troubleshoot > Akamai Diagnostics** window: **Akamai Connect License is Inactive. Please remove existing license and import new one using Akamai License page.**

Step 2 Remove the inactive or expired license.

Step 3 To upload a new license file, choose **Home > Admin > Licenses > Akamai Connect**.

Step 4 The **Akamai Connect** window is displayed.

Step 5 Click **Choose File** and browse to the new license file, and then click **Upload**.

If you try to import an expired license, you will see the message: **Unable to communicate to Akamai server (Error: License is inactive or expired). See Central Manager log file for detailed error information.**

Step 6 To obtain a new license, contact your Cisco account representative or reseller.

Enabling Akamai Connect

This section contains the following topics:

Confirming Akamai Connect Configuration Prerequisites

Before you enable Akamai Connect, confirm that your Cisco WAAS configuration has the following Akamai Connect prerequisites:

- **Cisco WAAS Version**: The Cisco WAAS Central Manager and Cisco WAAS appliances are running at Cisco WAAS Version 5.4.1 or later.

- **NTP Service**: A verified Network Time Protocol (NTP) service that is within 30 seconds of the NTP standard server (NTP.org). For more information, see Configuring NTP Settings in the chapter "Configuring Other System Settings."
- **DNS Server**: A working public Domain Name System (DNS) server configured on the Cisco WAAS devices and the Cisco WAAS Central Manager. For more information, see Configuring the DNS Server in the chapter "Configuring Network Settings."
- **Akamai Luna system and Akamai Management Gateway**: The ability for the Cisco WAAS Central Manager to reach Akamai's Luna system via HTTPS on port 443. (The custom hostname is in your activation file.)

  The ability for Cisco WAAS devices to make a connection to the Akamai Management Gateway (AMG) to get the authentication key. The Cisco WAAS device configured for Akamai Connect needs the correct network connectivity to access the AMG every day to get correct credentials and updated metadata. Cisco WAAS will make an HTTPS connection on port 443 to the AMG to get this information.

  Note The Akamai Connected Cache feature will stop functioning if Cisco WAAS loses communication with the AMG for more than 48 hours.

  If the Cisco WAAS devices cannot go directly to the Internet, you can configure them to use the Cisco WAAS Central Manager as a proxy. For more information, see Configuring the Cisco WAAS Central Manager as HTTP Proxy, on page 30.

Turning on the Akamai Cache Engine and Enabling Akamai Connect

Procedure

Step 1 From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure > Caching > Akamai Connect**.

The Akamai Connect window appears, with the **Cache Settings** tab displayed.

Note If you are configuring the Akamai Connect feature for a device group, the device group should have only devices that support Akamai Connect. For more information, see Supported Platforms for Cisco WAAS with Akamai Connect.

Step 2 To turn on the Akamai cache engine, check the **Enable Akamai Connect** check box.

The **End-User License Agreement - Akamai Connect** dialog box appears.

Step 3 Click **Accept**.

- When you create settings for the first time, either at the device or the group level, the **Akamai Connect Upload File** drop-down list is displayed. Choose the Akamai Connect license file and click **Submit**. For more information, see Activating and Managing the Akamai Connect License, on page 14.
- If you have not yet purchased an Akamai Connect license, see Activating and Managing the Akamai Connect License, on page 14.

Step 4 From the **Choose File** drop-down box, choose your Akamai Connect license file.

Step 5 Click **Submit** or proceed to Enabling Akamai Connected Cache.

Enabling Akamai Connected Cache

This section contains the following topics:

About Akamai Connected Cache

Akamai's proprietary caching rules in connection with the edge servers of the Akamai Intelligent Platform lets you cache and deliver content inside the branch office that might otherwise be deemed noncacheable. This content could be an enterprise's own web content, content that is served by the worldwide Akamai Content Delivery Network (Akamai CDN), or any content that is delivered by the Akamai Intelligent Platform, which is up to 30 percent of all web traffic.

Akamai Connected Cache contains the following features:

- Akamai Connected Cache is automatically enabled in Cisco WAAS when you enable Akamai Connect. You then specify the sites to be accelerated.
- Object caching is done on the client-side Cisco WAAS device only.
- Prepositioning can be leveraged to cache HTTP websites delivered via the Akamai Intelligent Platform.
- The Cisco WAAS/Akamai cache engine determines which sites can be "Akamaized" by Akamai Connected Cache from the HTTP headers in the first reply. The cache engine and the Akamai edge server then exchange credentials and agree that Akamai Connected Cache can occur. This is done again via HTTP headers in HTTP request and responses.
- During the enabling and registration of HTTP object cache, each Cisco WAE cache engine contacts the Akamai network to obtain credentials.

  After registration is complete, and Akamai Connected Cache is turned on, DNS requests are routed through the Akamai DNS system, and content is served up from an edge server to the Cisco WAAS router whenever it is possible.
- The Akamai edge server provides additional headers to allow the WAAS/Akamai cache engine to cache the objects for the objects it handles. The cache engine forwards this back to the corresponding client. The headers passed between the cache engine and the client are similar to what the client or enterprise proxy server would see if the Cisco WAE was not in the path.

Procedure for Enabling Akamai Connected Cache

Before you begin

For Akamai Connected Cache to function properly, you must have the following parameters configured:

- **Access to public DNS server**: For more information, see Configuring the DNS Server in the chapter "Configuring Network Settings."

- **NTP services**: For more information, see Configuring Date and Time Settings in the chapter "Configuring Other System Settings."

Procedure

Step 1 From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure > Caching > Akamai Connect**.

The **Akamai Connect** window appears, with the **Cache Settings** tab displayed.

Step 2 At the **Edit Settings** pane, check the **Akamai Connected Cache** check box. The default is **Enabled**.

Consider the following configuration guidelines for Akamai Connected Cache:

- When Akamai Connected Cache is enabled, it is enabled for all suitable Akamaized content.
- You can apply Akamai Connected Cache to a specified device, or to all registered devices:
  - To apply Akamai Connected Cache to all registered Cisco WAAS devices, configure Akamai Connected Cache at the device group level.
  - To apply Akamai Connected Cache to a specific registered Cisco WAAS device, configure Akamai Connected Cache at the device level.
- After you enable Akamai Connected Cache, you can perform the following tasks:
  - Set a caching policy for all sites, as described in Setting a Transparent Caching Policy for All Sites, on page 25.
  - Set an individual caching policy for a specific site, as described Setting a Transparent Caching Policy for a Specific Site, on page 26.
  - Enable Over the Top (OTT) caching, as described in Enabling Over the Top (OTT) Caching, on page 21.
  - Configure cache prepositioning, as described in Configuring Akamai Connect Cache Prepositioning, on page 35.

Step 3 Click Submit or proceed to Enabling Over the Top (OTT) Caching, on page 21.

Enabling Over the Top (OTT) Caching

This section contains the following topics:

About OTT Caching

Over-The-Top (OTT) caching is used for streamed content, particularly video content. OTT caching caches HTTP content served from dynamic URLs and content marked as noncacheable, such as YouTube videos. Akamai achieves this by using metadata logic to determine a unique cache key per video, which allows dynamic URLs to be cached. The following figure shows an example of OTT caching.

Note OTT caching is disabled by default. You can enable OTT caching after you enable Akamai Connected Cache. For more information, see Enabling Akamai Connected Cache, on page 20.

Sites that support OTT caching include the following:

- Apple
- Google
- Lynda
- Microsoft Updates
- Office 365
- Pearson
- Salesforce
- Schoology
- Vimeo
- Youku
- YouTube

Because YouTube is delivered via HTTPS, you must follow the same process as you do for Software as a Service (SaaS) optimization. The domains that must be matched are ***.youtube.com**, ***.ytimg.com**, ***.googlevideo.com**, and ***.ggpht.com**. For more information, see Configuring SSL Acceleration for SaaS Applications in the chapter "Configuring Application Acceleration."

Procedure for Enabling OTT Caching

Before you begin

Confirm that Akamai Connected Cache is enabled. For more information, see Enabling Akamai Connected Cache.

Procedure

Step 1 From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure > Caching > Akamai Connect**.

The **Akamai Connect** window appears, with the **Cache Settings** tab displayed.

Step 2 At the **Edit Settings** pane, check the **Over the Top Cache** check box.

Note You must enable Akamai Connected Cache before you enable OTT caching. For more information, see Enabling Akamai Connected Cache.

Step 3 Click **Submit** or proceed to tasks for setting caching policies: Setting a Transparent Caching Policy for All Sites or Setting a Transparent Caching Policy for a Specific Site.

Setting Transparent Caching Policies

This section contains the following topics:

Transparent Caching and Caching Modes

This section contains the following topics:

About Transparent Caching and HTTP Object Cache

Transparent cache is Akamai's high-performance HTTP object cache, which provides the ability to locally cache HTTP-based content for LAN-like performance, whether the web application was served from the private corporate cloud or from the public Internet. This content includes on-demand and live HTTP video streams, to deliver fast, high-quality, high-definition video in the branch, while offloading the enterprise network. Akamai Connect supports the latest generation of streaming protocols including Apple HTTP Live Streaming (HLS), Adobe HTTP Dynamic Streaming (HDS), and Microsoft HTTP Smooth Streaming (HSS). Akamai's HTTP object cache also supports the caching of Apple software updates such as iOS and OS X, and Microsoft Windows Update, further offloading the enterprise network.

Transparent caching delivers content from an origin server to the client without any modification. Transparent caching sends a request from a client to a server along with the associated authentication. No changes are made by proxy servers to either the headers or the returned packets along the way, although there are some headers that mark proxy actions that can be altered without the meaning of the cache control headers being altered.

Note When accessing transparent caching via HTTPS, the default caching mode is Basic mode. This ensures that no sensitive content is accidentally cached (in Basic mode, only content that you explicitly mark is cached). If you want content cached in a different mode with HTTPS, create a host rule that matches the HTTPS server location. For more information on creating a host rule, see Setting a Transparent Caching Policy for All Sites and Setting a Transparent Caching Policy for a Specific Site.

Transparent caching modes are used to set caching policies. For more information, see Transparent Caching Modes, on page 23.

Transparent Caching Modes

There are four types of Transparent caching modes, or policies: Basic, Standard (default), Advanced, and Bypass.

- **Basic Transparent Caching Mode**

  Basic mode is the lowest level of caching, where it strictly complies with the client caching directives in the HTTP header, caching only objects marked explicitly as cacheable. Caching is only in the branch or local router, and content can be cached from the Internet regardless of the location of the original source.

- **Standard Transparent Caching Mode (default)**

  Standard (default) caching mode expands the breadth of caching objects by including objects marked as cacheable, objects that do not have caching directives, and with a last-modified date. For example, with Standard caching, the object will be cached for 10 percent of the current age of the response and then updated.

  Note A correctly configured website will work with Standard mode, but login pages, cookie setting pages, or dynamic content not properly marked as cacheable may break. We recommend that you test the website; this is particularly important for a newly-created website or one that does not have many users.

- **Advanced Transparent Caching mode**

  Advanced caching mode further extends the duration for which the objects without specific age limits are cached, thus allowing an aggressive amount of caching in appropriate situations, and to cache all object types for longer times, when there is no explicit expiration time. Advanced mode is best suited for media-rich Intranet sites.

  If cache-control or expire headers are not present and **Last Modified Time** appears, the cache engine performs a heuristic based on the **Last Modified Time** and stores objects for 20 percent of their apparent age, up to a maximum of one day.

  For certain media file types, listed in the table "Advanced Mode: Media Types That May be Cached for a Full Day," Advanced Mode will cache these for a full day if the media type is not specified as uncacheable or the media type has no obvious age in the request. For all other media types, the system caches the object for a minimum of one hour to a maximum of seven days - regardless of whether the **Last Modified Time** is present.

Table 7: Advanced Mode: Media Types That May be Cached for a Full Day

Media Types That May be Cached for a Full Day, if not specified as uncacheable or with no obvious age in request										
3g2	3gp	aac	aif	aiff	asf	asx	au	avi	bin	bmp
cab	carb	cct	cdf	class	css	dcr	doc	docx	dtd	dv
dvd	dvr	dvr-ms	exe	flv	gcf	gff	gif	grv	hdml	hqx
ico	ini	jpeg	jpg	js	mlv	m4a	midi	mov	mp3	mp4
mpeg	mpg	mpv	nv	pct	pdf	png	ppc	ppt	pptx	pws
qt	swa	swf	tif	txt	vbs	w32	wav	wbmp	wma	wml
wmlc	wmls	wmlsc	wmv	xsd	xsl	xls	xlsx	zip	–-	–-

Note A correctly configured website will work in Advanced mode, but Advanced mode may break the presentation of certain web pages if there are even minor caching misconfigurations. We recommend that you test the performance of this caching mode for your applications before you bring the cache engine into production. When testing, pay particular attention to dynamic URLs and to content that requires authentication to be presented to a client.

- **Bypass Transparent Caching Mode**

  Bypass mode turns off caching for a configured site or sites. When Bypass mode is set for a particular hostname, the caching for the site's hostname specified in a rule is suppressed.

  Bypass mode is useful when you want to turn off Akamai Connected Cache or OTT caching for a site or for a part of a site. For example, if you have servers of the type **images#.bar.com**, you can configure a bypass rule so that only **images2.bar.com** is excluded from caching. All other **images#.bar.com** servers will continue to be cached under the existing rules.

Order of Preference for Caching Types

Consider the following operating guidelines for order of preference for caching types:

- When there are multiple caching modes and policies in use, the Cache Engine applies an order of precedence in the execution of these. A rule that is higher in the order of precedence is executed first, and any other rules that are applied to that domain or digital property is ignored. The order of preference is:

  1. Transparent caching rules
  2. OTT/Akamai Connected Cache
  3. Default Transparent caching policy

  For example, if **test.com** is an Akamai Connected Cache property, but an Advanced mode cache rule is set for this site, then Advanced mode will take precedence and Akamai Connected Cache will be skipped.

- When cache prepositioning is turned on, it has the same priority as any other caching type.
- Akamai Connect determines cache type based on most exact hostname match followed by cache priorities. For example, **www.host.com** is more exact than ***.host.com**. In this scenario, if a lower-priority cache, such as Akamai Connected Cache (Order of Precedence #2), has a more exact match than a higher priority cache, such as transparent (Order of Precedence #1), the caching will occur with the more exact match and lower-priority cache.

Setting a Transparent Caching Policy for All Sites

Procedure

Step 1 From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure > Caching > Akamai Connect**.

The **Akamai Connect window** appears, with the **Cache Settings** tab displayed.

Step 2 At the Advanced Cache Settings pane, from the Default Transparent Caching Policy drop-down list, choose one of the following caching policies as a default transparent caching policy for all sites:

- **Basic**: Caches only objects marked explicitly as cacheable.
- **Standard (default)**: Caches objects marked as cacheable, as well as objects that do not have caching directives or a last-modified date.
- **Advanced**: Further extends the duration for which the objects without specific age limits are cached, thus allowing an aggressive amount of caching in appropriate situations, and to cache all object types for longer times, when there is no explicit expiration time.
- **Bypass**: Turns off caching for a specific configured site or sites.

Considering the following about caching polices:

- Checking the Akamai Connected Cache check box () starts active caching with the default Standard caching policy. To use the Akamai cache engine with a Basic, Advanced, or Bypass caching policy, you must specify that caching policy with the Default Transparent Caching Policy drop-down list.
- To set a caching policy for a specific site, see Setting a Transparent Caching Policy for a Specific Site.
- For more information about caching policies, see Transparent Caching and Caching Modes, on page 23.

Step 3 Click **Submit** or proceed to Setting a Transparent Caching Policy for a Specific Site.

Setting a Transparent Caching Policy for a Specific Site

Procedure

Step 1 From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure** > **Caching** > **Akamai Connect**.

The **Akamai Connect** window appears, with the **Cache Settings** tab displayed.

Step 2 At the **Advanced Cache Settings** pane, from the **Default Transparent Caching Policy** drop-down list, choose **Bypass**.

Choosing **Bypass** turns off caching, so that you can set a specific caching policy for the site.

Step 3 To add a site to contain a specific caching policy, at the **Site Specific Transparent Caching Policy** table listing, click **Add Hostname/IP**.

The **Site Caching Policy Task** dialog box appears.

Step 4 In the **Site Caching Policy Task** dialog box, in the **Hostname/IP** field, specify the hostname of the site to be configured.

Consider the following guidelines for creating a hostname:

- The hostname can be a specific server, or a domain name that contains a wildcard, such as ***.cisco.com**.

- You can configure up to 512 hostnames for each site-specific caching policy.
- From the **Transparent Caching Policy** drop-down list, select the cache policy for this site: **Basic**, **Standard**, **Advanced**, or **Bypass**.
- Consider the following guidelines for setting a site-specific cache policy:
  - The policy you set for a specific site takes precedence over the default caching policy set for all sites.
  - If you configure **Bypass** mode as the site-specific transparent caching policy, you must specify a complete server name or a complete domain name (a Fully Qualified Domain Name [FQDN]). If you use a wildcard to specify sites for **Bypass** mode, the sites will still be optimized via Akamai Cache.

Step 5 Click **OK**.

The new hostname/IP is added as a line item to the **Site Specific Transparent Caching Policy** table.

- To edit an existing site, highlight the site listing and click **Edit**.
- To delete an existing site, highlight the site listing and click **Delete**.

Step 6 Click **Submit** or proceed to Enabling Cisco Cloud Web Security (Cisco CWS).

Enabling Cisco Cloud Web Security (Cisco CWS)

This section contains the following topics:

About Cisco CWS

Cisco Cloud Web Security (CWS) provides content scanning of HTTP and HTTP/S traffic, and provides malware protection service to web traffic. CWS enforces content filtering by enabling force IMS for every cached object, for both single-sided and dual-sided deployment.

CWS servers scan web traffic content and either allow or block the traffic based on configured policies. Servers use credentials to identify and authenticate users and redirect the traffic for content scanning. Traffic is transparently proxyed by Cisco routers to cloud-based CWS servers, where the web traffic is scanned and, if deemed acceptable, is provided to the origin server. All traffic coming back is through the CWS server.

Cisco CWS Operating Guidelines

Consider the following operating guidelines for Cisco CWS:

- Cisco CWS version interoperability:
  - For Cisco WAAS Version 6.2.1 and later, the CWS feature enforces content filtering by enabling force IMS for every cached object, for both single-sided and dual-sided deployment.
  - For Cisco WAAS Versions earlier than 6.2.1, content filtering is enforced on single-sided deployments.

- CWS can be used only when one Cisco WAAS device is present in the path.
- If preposition is enabled, the traffic flow may be redirected to a CWS server, follow these recommendations:
  - (Preferred choice): Configure a white list on the Cisco ISR or the Cisco CWS server to bypass the Cisco WAE IP address.
  - On the Cisco CWS server, configure a user or group that the Cisco WAE will fall into for authentication and allow it access to all sites on which the preposition is occurring.

Procedure for Enabling Cisco CWS

Procedure

Step 1 From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure** > **Caching** > **Akamai Connect**.

The **Akamai Connect** window appears, with the Cache Settings tab displayed.

Step 2 At the **Advanced Cache Settings** pane, to enable Cisco CWS:

To enable CWS user policy enforcement for content access with Direct Internet Access (DIA), check the **Force IMS DIA** check box.

To apply CWS user policy enforcement for content access with all flows, check the **Force IMS Always** check box.

Step 3 Click **Submit**, or proceed to Configuring Cisco WAAS Connections to the Akamai Network, on page 28.

Configuring Cisco WAAS Connections to the Akamai Network

This section contains the following topics:

About Cisco WAAS Connections to the Akamai Network

This section provides an overview of the three ways for Cisco WAAS devices to connect to the Akamai network.

- Configure no HTTP proxy.
- Configure the Cisco WAAS Central Manager as HTTP proxy.
- Configure an external HTTP proxy.

When using Akamai Connect, the Cisco WAAS Central Manager and Cisco WAAS device(s) must be able to communicate with the Akamai Network: with the Akamai Luna API servers to provision entries for Cisco WAAS devices, and with the Akamai AMG devices for Akamai Connected Cache and OTT features.

However, some Cisco WAAS deployments may disallow outgoing connections to the Internet for the Cisco WAAS Central Manager or Cisco WAAS device(s). For these deployments, the Cisco WAAS device(s) may use an HTTP proxy to contact the Akamai Network.

Note HTTP proxy must support **HTTP CONNECT** for tunneling HTTPS connections.

The following table shows the available connection configurations.

Table 8: Connection Configurations for Cisco WAAS to Akamai Network

Connection Configuration	Configuration Connections	Cisco WAAS Central Manager to Luna API Servers	Cisco WAAS HTTP Cache Engine to Akamai AMG
No HTTP proxy use	Direct/ Direct	Direct	Direct
Cisco WAAS Central Manager as HTTP proxy	Direct/ Cisco WAAS Central Manager as proxy	Direct	Cisco WAAS Central Manager as HTTP proxy
External HTTP proxy	Direct/ External HTTP proxy	Direct	External HTTP proxy
External HTTP proxy	External HTTP proxy/ Direct	External HTTP proxy	Direct
External HTTP proxy	External HTTP proxy/ External HTTP proxy	External HTTP proxy	External HTTP proxy

The following considerations apply to all HTTP proxy deployments:

- You configure HTTP proxy from the Cisco WAAS Central Manager; there are no CLI commands for HTTP proxy. Configuring HTTP proxy settings does not require restart of the WAAS Central Manager.
- HTTP Proxy must support HTTP Connect method for tunneling HTTPS connections.
- Configuring the HTTP proxy setting does not require restart of the Cisco WAAS Central Manager.

Note Cisco WAAS v5.5.1 does not support HTTP proxy user authentication. We recommend that you restrict access to proxy using IP address ACLs.

Configuring No HTTP Proxy

Procedure

To configure a direct connection from the Cisco WAAS Central Manager and Cisco WAAS devices to the Akamai network: From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure** > **Caching** > **Akamai Connect**.

The Akamai Connect window appears, with the **Cache Settings** tab displayed.

a) At the **Advanced Cache Settings** pane, confirm that the **Use HTTP proxy for connections to Akamai network** check box is unchecked.

a) Click **Submit**.

Configuring the Cisco WAAS Central Manager as HTTP Proxy

This section contains the following topics:

Operating Guidelines for Cisco WAAS Central Manager as HTTP Proxy

Consider the following operating guidelines when using the Cisco WAAS Central Manager as an HTTP proxy to the Akamai network:

- When using Akamai Connected Cache, each Cisco WAAS cache engine device is communicating with the Akamai network. Some Cisco WAAS deployments may disallow WAE devices to establish outgoing connections to the Internet (i.e., private networks). In this case, the WAE device may use the Cisco WAAS Central Manager device(s) as proxy for all connections to the Akamai network.
- You may still have to allow a hole for the Cisco WAAS Central Manager to make communications on TCP port 443 outbound.
- There is no option for the Cisco WAAS Central Manager to use a proxy device to get to the Internet.
- All connections are made from the Cisco WAAS cache engine device or Cisco WAAS Central Manager out to the Akamai network; never from the Akamai network to the Cisco WAAS cache engine device or Cisco WAAS Central Manager.
- You configure this feature from the Cisco WAAS Central Manager only, not from the Cisco WAAS CLI.

Procedure for Configuring the Cisco WAAS Central Manager as HTTP Proxy

Procedure

Step 1 From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure** > **Caching** > **Akamai Connect**.

The **Akamai Connect** window appears, with the **Cache Settings** tab displayed.

Step 2 At the **Advanced Cache Settings** pane, check the **Use HTTP proxy for connections to Akamai network** check box.

Step 3 From the **HTTP Proxy**: drop-down list, choose **Central Manager as HTTP Proxy**.

Step 4 Click **Submit**.

Configuring External HTTP Proxy

This section contains the following topics:

About External HTTP Proxy

When using Akamai Connect, some Cisco WAAS deployments may disallow outgoing connections to the Internet for the Cisco WAAS Central Manager or Cisco WAAS device(s). For these deployments, the Cisco WAAS device(s) may use an HTTP proxy to contact the Akamai Network.

Configuring External HTTP Proxy for a Device or Device Group

Procedure

Step 1 From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure > Caching > Akamai Connect**.

The **Akamai Connect** window appears, with the **Cache Settings** tab displayed.

Step 2 Check the **Use HTTP proxy for connections to Akamai network** check box.

Step 3 At the **Advanced Cache Settings** pane, from the **HTTP Proxy**: drop-down list, choose **External HTTP Proxy**.

Step 4 Specify a **Proxy Host** and a **Proxy Port**:

Proxy Host field: Enter a hostname or IP address.

Proxy Port field: Enter a value between **1** to **65535**.

If the Cisco WAAS Central Manager is already using an external HTTP proxy, there is no option displayed to use the Cisco WAAS Central Manager as proxy; these fields will display the currently configured HTTP proxy.

Step 5 Click **Submit**.

Configuring External HTTP Proxy for All Devices

Procedure

Step 1 From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure > Global > External HTTP Proxy**.

The following message is displayed:

Some deployments may disallow direct connections from Central Manager to Internet hosts. This would affect WAAS features such as Akamai Connect, where Central Manager needs to communicate with Akamai servers. For such deployments WAAS Central Manager may use an external HTTP proxy to contact Internet. HTTP proxy must support HTTP CONNECT method for tunneling HTTPS connections.

Step 2 Specify a **Proxy Host** and a **Proxy Port**:

Proxy Host field: Enter a hostname or IP address.

Proxy Port field: Enter a value between **1** to **65535**.

If the Cisco WAAS Central Manager is already using an external HTTP proxy, there is no option displayed to use the Cisco WAAS Central Manager as proxy; these fields will display the currently configured HTTP proxy.

Step 3 Click **Submit**.

Configuring Server Address Validation

This section contains the following topics:

About Server Address Validation

Server Address Validation prevents malicious content from infecting the Akamai Connect cache, by performing Domain Name Service (DNS) lookups on the name in the HTTP host header, comparing the lookup result with that connection's forward IP address, and, if there is a mismatch, the transaction is allowed to pass through the cache, but no content is allowed to be cached.

Server Address Validation is available for Cisco WAAS Version 6.4.1 and later.

Example:

1. The server IP address to which a DNS name resolves might be an IP address of a server that contains malicious content, rather than that of an expected and trusted server.
2. The resulting response would get cached, and the Akamai cache would then contain malicious content.
3. After the cache is "poisoned" with this malicious content, other clients accessing the same content would also get served with this malicious data.

To prevent such situations, Server Address Validation provides the following features:

- Performs DNS lookups on the name in the HTTP host header.

  A valid Domain Name System (DNS) configuration is required for Server Address Validation to work properly. For more information, see Configuring the DNS Server in the chapter "Configuring Network Settings."

- Compares the lookup result with that connection's forward IP address.
- If there is a mismatch, the transaction is allowed to pass through the cache, however, no content is allowed to be cached.

Note The Cisco Cloud Web Security (Cisco CWS) feature also performs traffic scanning and malware protection for the Akamai Connect cache. For more information on Cisco CWS, see Enabling Cisco Cloud Web Security (Cisco CWS).

Alarms Used with Server Address Validation

The following table shows the alarms used with Server Address Validation.

Table 9: Alarms Used with Akamai Connect Cache Server IP Address Validation

Alarm Name	Reason Alarm is Raised	User Action
DNS Lookup Failed	Too frequently, address checks have been unable to look up hostnames.	Verify that the DNS configured in Cisco WAAS is able to do host lookup.
Forward Proxy Detected Warning	Address checks are enabled, but a forward proxy that is re-looking up hostnames has been detected.	Do one of the following: • Disable address checks. • Disable the forward proxy from re-looking up hostnames. • Disable address checks for the forward proxies' IP addresses. • Add forward proxies' IP addresses to the whitelist.
Address Check Failures Warning	Server address validation has found that too many address mismatches, the IP address returned by DNS lookup does not contain the address used by the client, are occurring.	Use the **show hosts** EXEC command to list and verify the name servers and their corresponding IP addresses, and to list the hostnames, their corresponding IP addresses, and their corresponding aliases (if applicable).

Procedure for Configuring Server Address Validation

Before you begin

This section describes how to use the Cisco WAAS Central Manager to enable or disable Server Address Validation, and to add, edit, or delete bypass server addresses into or from a whitelist.

Before you configure Server Address Validation, consider these guidelines:

- A valid Domain Name System (DNS) configuration is required for Server Address Validation to work properly. For more information, see Configuring the DNS Server in the chapter "Configuring Network Settings."
- If Interposer-SSL is disabled, the following warning message is displayed:

  Interposer-SSL is in disabled state. Enable Interposer-SSL for HTTP Object Cache Server Validation feature to use SNI extension. Peformance for HTTPS connections, when this feature is enabled, might get affected in the absence of SNI.

  For more information, see Enabling and Disabling Global Optimization Features in the chapter "Configuring Application Acceleration."

- To configure Server Address Validation with the Cisco WAAS CLI, run the **accelerator http object-cache validate-address enable** and **accelerator http object-cache validate-address bypass** global configuration commands. For more information, see the *Cisco Wide Area Application Services Command Reference*.

Procedure

Step 1 To enable or disable Server Address Validation:

a) From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure** > **Caching** > **Akamai Connect**.

The **Akamai Connect** window appears, with the **Cache Settings** tab displayed.

b) At the **Server Address Validation** pane, check the **Enable server address validation** check box. The default is disabled.

Consider the following operating guidelines for enabling or disabling Server Address Validation:

- If Server Address Validation is enabled or disabled at the device group level, the option is enabled or disabled to all devices in the device group.
- If you have enabled or disabled Server Address Validation from the Cisco WAAS CLI, it will be reflected in the Cisco WAAS Central Manager within two data feed cycles.

Step 2 To create a server address whitelist:

a) At the **Bypass Server Address** table listing taskbar, click **Add Server IP Address**.

The **Bypass Server** dialog box appears.

b) In the **Bypass Server IP** field, specify the server IP address.
c) In the **Netmask** field, specify the netmask.
d) Click **OK**.

The new server IP address and netmask are added to the **Bypass Server Address** table listing.

- You can configure up to 50 server IP addresses per whitelist.
- To edit an existing bypass server address, highlight the bypass server address and click **Edit**.
- To delete an existing bypass server address, highlight the bypass server address and click **Delete**.

Note A server address whitelist that you have created is stored on the data server until you delete it. The server IP address whitelist is *not* automatically deleted if you disable Server Address Validation.

Upgrade and Downgrade Considerations for Server Address Validation

- Downgrading to a Cisco WAAS version earlier than WAAS Version 6.4.1: Server Address Validation is not available for Cisco WAAS versions earlier than 6.4.1.
- Upgrading from Cisco WAAS Version 6.4.1 to a later version: Server Address Validation is available for all Cisco WAAS versions 6.4.1 and later.

Configuring Akamai Connect Cache Prepositioning

This section contains the following topics:

About Akamai Connect Cache Prepositioning

Cache prepositioning, also known as cache warming, allows you to specify a policy to prefetch and cache content at a specified time. Cache prepositioning allows you to take advantage of idle time on the WAN to transfer large or frequently accessed files to selected Cisco WAAS devices, so that users can benefit from cache-level performance even during first-time access of these files.

Cache prepositioning fetches content based on:

- Predefined schedule
- URL and link depth level
- Excluded content types

Cache prepositioning runs at the same priority as other caching types, for example, Akamai Connected Cache or OTT.

For Cisco WAAS Version 6.2.1 and later with Akamai Connect, cache prepositioning for Akamai Connect also provides the following cache prepositioning features:

- Processing of manifest files for the video streaming protocols HLS (HTTP Live Streaming) and HDS (HTTP Dynamic Streaming).
- Prepositioning of JNLP (Java Network Launch Protocol) files, which contain URL reference for Java Web Start.

Operating Guidelines for Akamai Connect Cache Prepositioning

Consider the following operating guidelines for cache prepositioning for Akamai Connect:

- When a scheduled fetch operation begins or is complete, it is added to the Cache Preposition Status table.
- In order for HTTP/S content to be prepositioned, you must define an SSL accelerated service; otherwise, any HTTP requests encountered in the job will fail, although the preposition task will continue and any objects available via HTTP will be retrieved.

  For more information on how to define an SSL accelerated service, see Configuring SSL Acceleration in the chapter "Configuring Application Acceleration."

Configuring a Cache Prepositioning Task

Before you begin

The following table shows the dialog boxes, available from the **Cache Prepositioning** tab, used to configure a preposition task.

Table 10: Cache Prepositioning Dialog Boxes

Dialog Box	Description
Cache Prepositioning Task	Used to specify the preposition task name, base URLs for prepositioning, include and exclude types, download rate, recursion depth, and task duration.
Cache Prepositioning Task, Advanced Settings	Used to specify the recursion delay time and recursion domains.
Cache Prepositioning Schedule	Used to specify the schedule name for the preposition task, frequency of the task (such as daily or monthly), and start time.

Procedure

Step 1 From **Devices** or **Device Groups**, choose **Configure > Caching > Akamai Connect**.

The **Akamai Connect** window appears, with two tabs: **Cache Settings** and **Cache Prepositioning**.

Step 2 Choose the **Cache Prepositioning** tab.

At this tab, you can add, edit, or delete cache prepositioning tasks, as well as monitor cache preposition task status.

Step 3 (Optional) To enable DRE for preposition connections, check the Preposition with DRE check box. The default is disabled, to prevent negative impact to the DRE byte cache for data that will be stored at the object level.

Step 4 At the **Cache Prepositioning** table listing, click **Add Cache Preposition Task**.

The **Cache Prepositioning Task** dialog box appears.

Step 5 In the **Name** field, enter the name of the preposition task.

- Preposition task name is an alphanumeric identifier up to 47 characters. Special characters like ‘,/,\,{,},(,),?,",<,>,[,],&,*, are not allowed.
- You can configure up to 10 URLs per task.
- You can configure up to 10 schedules per task.
- You can configure up to 50 tasks per device or device group.

Step 6 In the **URLs** field, enter the base URLs for prepositioning.

- The maximum length for the URL is 900 characters. Characters that are not allowed in the URL are space, double quotes (“). ASCII characters are allowed in the range of ASCII 33 through ASCII 125.
- Use a space to separate multiple URLs.
- You can configure up to 10 URLs per task.

Step 7 In the **Exclude Types** field, enter the object types to exclude from caching, such as .jsp or .asp, each separated by a comma.

The list of object name patterns to be excluded has a total pattern field limit of 47 characters.

Step 8 In the **Download Rate** field, enter the maximum download rate, in KBps. Select any value between 0 to 10,000,000 KBps.

- The default is 20 KBps.
- A selection of 0 indicates unlimited, or no enforced rate limiting.

Step 9 To enable recursion for this cache preposition task, check the Recursive Task check box. To have recursion disabled for this cache preposition task, leave the Recursive Task check box unchecked. The default is unchecked.

Step 10 If you have checked the **Recursive Task** check box, from the **Recursion Depth** drop-down list, choose the depth of the link level at which content is retrieved: 1, 2, 3, 5, 8, 13, or 21. You can also enter a custom value from 1 to 1000. The default recursion depth value is 1.

The **Recursion Depth** drop-down list is active only if you check the **Recursive Task** check box.

Note A greater number of specified levels of links means a greater amount of data stored in the cache, sometimes exponentially more. If the amount of requested prefetched data becomes larger than the cache, the newly requested data will flush all previously stored data, and may slow down other operations that attempt to use the cache.

Step 11 To enable this cache preposition task, check the **Enable Task** check box. The task must specify at least one URL (specified in the URLs field) and one schedule, specified in the next step.

Step 12 At the **Cache Prepositioning Schedule** table listing, click Add Schedule.

The **Cache Prepositioning Schedule** dialog box appears.

a) In the **Schedule Name** field, enter the name of the schedule of this cache preposition task, up to 256 alphanumeric characters. The schedule name allows you to provide your own representation of a schedule. For example, you can name a schedule that occurs every Monday, Wednesday, and Friday at 10:30 a.m. as **Weekly MWF 10:30AM** or as **Every Week - Mon-Wed-Fri at 10:30AM**.

b) From the **Frequency** drop-down list, choose the specified time for prepositioning: yearly, daily, weekly, or monthly days.

For example, if you choose you choose monthly days, a calendar with check boxes opens for you to check one, some, or all the days in a month for this schedule.

c) From the **Start Time (HH:MM)** drop-down lists, choose the hour and minute at which this cache prepositioning task should start.

d) Click **OK**.

Step 13 At the **Advanced Settings** section of the **Cache Prepositioning Task** dialog box, you can specify recursion delay time and recursion hostnames.

In the **Recursion Delay Time** field, enter the delay time, in seconds, between requests during recursive download. This simulates user wait time. Recursive delay time is necessary because some servers use the lack of time between requests to detect and restrict web crawlers.

- Enter a value between 0 to 600 seconds. The default is 2 seconds.
- A value of zero provides the best performance when there are no web crawler restrictions.

Step 14 In the **Recursion Domains** field, enter the list of server domain suffixes for which recursive web crawling is permitted. If this list is empty, then web crawling is only permitted within the same domain as the specified URL.

You can configure up to ten servers:

- The server name is up to 255 alphanumeric characters.
- Server names are separated by comma or space.

Step 15 Click **OK**.

Step 16 In the **Cache Prepositioning Schedule** dialog box, click **OK**.

Step 17 In the **Cache Prepositioning Task** dialog box, click **OK**.

Step 18 Click **Submit**.

The new cache prepositioning task is added as a line item in the **Cache Prepositioning** table listing.

Viewing Cache Prepositioning Task Status

The **Cache Prepositioning** pane provides two tables to show the status of a cache preposition task.

- To view the status of a cache preposition task you have configured, highlight and select the task from the first table, the **Cache Prepositioning** table listing.
- The second table, the **Cache Prepositioning Status** table listing, displays information on the selected task.
  - For an individual device: The cache prepositioning status table shows the selected task status for the current device.
  - For a device group: The cache prepositioning status table shows the status of the selected cache preposition task, for all devices under that device group.

The following table displays information for the selected cache prepositioning task.

Table 11: Cache Prepositioning Task Status Details

Cache Prepositioning Task Status Table Column	Description
Device Name	The name of the selected device.
Start Time	The date, hour, and minute for the task schedule to start.
End Time	The date, hour, and minute for the task schedule to end.
Byte Count	The total number of bytes in cache during the most recent preposition task run.
Object Count	The total count of objects in cache during the most recent preposition task run.

Cache Prepositioning Task Status Table Column	Description
Refresh Bytes	The number of bytes refreshed in cache during the most recent preposition task run.
Refresh Count	The count of objects refreshed in cache during the most recent preposition task run.
Store Bytes	The number of unmodified bytes for objects found in cache during the most recent task run.
Store Count	The count of unmodified objects found in cache during the most recent task run.
Uncacheable Bytes	The number of bytes of uncacheable objects encountered during the most recent task run.
Uncacheable Count	The count of uncacheable objects encountered during the most recent task run.
Status	The status of the task, such as **Scheduled**, **Complete**, or **Error**.
Error	If the task status is **Error**, an error message describing the task status is displayed.

Copying Cache Prepositioning Tasks

Before you begin

You can copy cache prepositioning tasks that have a device or device group enabled with Akamai Connect. Use the following methods to copy cache prepositioning tasks:

- Device to device
- Device to device group
- Device group to device
- Device group to device group

Procedure

Step 1 From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure** > **Caching** > **Akamai Connect**.

The **Akamai Connect** window appears, with two tabs: **Cache Settings** and **Cache Prepositioning**.

Step 2 Choose the **Cache Prepositioning** tab.

Step 3 At the **Cache Prepositioning** pane, click **Copy Tasks**.

Step 4 From the **From** drop-down list, choose a device or device group as the source.

Step 5 From the **To** drop-down list, choose a device or device group as the destination.

Note If you try to copy a task with the same name between device and device groups, the following error message is displayed: **One or more preposition tasks with the same name already exists in the destination device/DG**.

Step 6 At the **Existing Cache Prepositioning Tasks** table listing, select one, some or all of the cache preposition tasks to be copied.

Step 7 Click **OK**.

The selected cache prepositioning tasks are copied from the specified source to the specified destination.

Configuring HTTP/S Preposition Proxy for Akamai Connect

This section contains the following topics:

About HTTP/S Preposition Proxy for Akamai Connect

For Cisco WAAS Version 6.2.1 and later, you can preposition external content in the case of a deployment with proxy. Consider the following when configuring HTTP/S preposition proxy for Akamai Connect:

- IPv4 proxy is supported for HTTP/S prepositioning.
- The HTTP preposition proxy feature is a feature independent of the Cisco WAAS Central Manager and external HTTP proxy features described in the sections Configuring the Cisco WAAS Central Manager as HTTP Proxy and Configuring External HTTP Proxy.
- Specific IP address-based proxy configuration is supported for HTTP/S preposition proxy. File-based and auto-detected configurations are not supported for HTTP/S preposition proxy.

Configuring Global Proxy Host and Port for Preposition Tasks

Procedure

Step 1 From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure > Caching > Akamai Connect**.

The **Akamai Connect** window appears, with two tabs: **Cache Settings** and **Cache Prepositioning**.

Step 2 Choose the **Cache Prepositioning** tab.

Step 3 In the **Proxy Host** field, enter the hostname or IP address for the proxy host.

Step 4 In the **Proxy Port** field, enter the port number. Valid port numbers are 0 to 65535.

Step 5 Click **Submit**.

Step 6 Create a preposition task, as described in Configuring a Cache Prepositioning Task, on page 35.

Step 7 In the **Cache Prepositioning Task** dialog box, check the **Enable Proxy** check box.

Step 8 Schedule the task, as described in Step 12 of Configuring a Cache Prepositioning Task.

Step 9 Click **Submit**.

Modifying Proxy Settings for an Individual Prepositioning Task

Procedure

Step 1 From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure** > **Caching** > **Akamai Connect**.

The **Akamai Connect** window appears, with two tabs: **Cache Settings** and **Cache Prepositioning**.

Step 2 Choose the **Cache Prepositioning** tab.

Step 3 Select a cache prepositioning task that you have configured as proxy.

Step 4 Modify the particular setting or settings.

Step 5 Check the **Enable Task** check box.

Step 6 Check the **Enable Proxy** check box.

Step 7 In the **Cache Prepositioning Schedule** dialog box, select parameters to reschedule the task.

Step 8 Click **OK**.

Step 9 In the **Cache Prepositioning Task** dialog box, click **OK**.

Step 10 Click **Submit**.

Removing Proxy Settings for an Individual Prepositioning Task

Procedure

Step 1 From the Cisco WAAS Central Manager menu, from either the **Device Groups** or **Devices** tab, choose **Configure** > **Caching** > **Akamai Connect**.

The **Akamai Connect** window appears, with two tabs: **Cache Settings** and **Cache Prepositioning**.

Step 2 Choose the **Cache Prepositioning** tab.

Step 3 Select a cache prepositioning task that you have configured as proxy.

Step 4 Check the **Enable Task** check box.

Step 5 Uncheck the **Enable Proxy** check box.

Step 6 In the **Cache Prepositioning Schedule** dialog box, select parameters to reschedule the task.

Step 7 Click **OK**.

Step 8 In the **Cache Prepositioning Task** dialog box, click **OK**.

Step 9 Click **Submit**.

Cisco Support for Microsoft Windows Update

Cisco support for Microsoft Windows Update enables caching of objects used in Windows OS and application updates. Cisco support for Microsoft Windows Update is enabled by default, and enabled only for specific sites.

This section contains the following topics:

About Cisco Support for Microsoft Windows Update

The Microsoft operating system and application updates are managed by update clients such as Microsoft Update. Microsoft Update downloads the updates via HTTP, often in combination with BITS (Background Intelligent Transfer Service) to help facilitate the downloads. Clients use HTTP range request to fetch updates.

The objects that comprise the updates, such as .cab files, are typically cacheable, so that HTTP object cache is a significant benefit for this process.

For example, for Windows 7 and 8 OS updates, via direct Internet or WSUS (Windows Server Update Services), versions 2012 and 2012R2, more than 98% of the update files, such as .cab, .exe, and .psf files, are served from cache on subsequent updates. Cisco support for Microsoft Windows Update reduces the volume of WAN offload bytes and reduces response time for subsequent Windows updates.

Viewing Statistics for Cisco Support for Microsoft Windows Update

There are two ways to view data generated by Cisco support for Microsoft Windows Update:

- Akamai Connected Cache Charts in the chapter "Monitoring Your Cisco WAAS Network," provides information including WAN response time and WAN offload bytes.
- For Cisco WAAS Version 6.1.1 and later, the cache engine access log file has two new fields for Microsoft Windows Update statistics:
  - **rm-w** (range miss, wait): The main transaction, a cache miss, which waited for the sub-transaction to fetch the needed bytes

    .
  - **rm-f** (range miss, full): The sub-transaction, a cache write of the entire document.

Example 1:

Example 1 contains two log lines, the main transaction and sub-transaction, when a range is requested on an object that is not in cache:

```
ws8-rt-kb2863725-x64_dd8522e527483cd69bf61d98ee849a2406b97172.psf - -
08/28/2015 12:22:29.663 (fl=27520) 300 13.164 0.000 446 - - 34912 172.25.30.4

191.234.4.50 2905 h - - rm-w 206 GET
http://fg.v4.download.windowsupdate.com/d/msdownload/update/software/secu/2013/07/wind
ows8-rt-kb2863725- x64_dd8522e527483cd69bf61d98ee849a2406b97172.psf - -

08/28/2015 12:24:31.448 (fl=27520) 300 134.949 0.000 355 344 3591542 568 172.25.30.4
191.234.4.50 2f25 m-s - - rm-f 200 GET
http://fg.v4.download.windowsupdate.com/d/msdownload/update/software/secu/2013/07/wind
ows8-rt-kb2863725-x64_dd8522e527483cd69bf61d98ee849a2406b97172.psf - -
```

Example 2:

Example 2 shows a cache hit when a range is requested on an object that is either completely in cache, or in the process of being downloaded. If it is in the process of being downloaded, then the main transaction has latched onto a sub-transaction like the one shown in Example 1.

```
08/28/2015 03:34:36.906 (fl=26032) 300 0.000 50.373 346 - - 13169 172.25.30.4
8.254.217.62 2905 h - - - 206 GET
http://fg.v4.download.windowsupdate.com/d/msdownload/update/software/secu/2013/07/wind
ows8-\ rt-kb2863725-x64_dd8522e527483cd69bf61d98ee849a2406b97172.psf - -
```

Cisco Support for Microsoft Windows Update and Akamai Cache Engine

Cisco support for Microsoft Windows Update enables Akamai Cache Engine to support Windows Update caching in two ways:

- Download and cache full objects even when ranges within objects that not in cache are requested.
- Future range requests on the objects can be served out of cache.

There is a limit, set by OTT metadata during the Akamai Connect registration process, from the start of the object - the number of bytes or the percent of file length - where the download functionality is triggered. A request of a size above the set limit does not initiate a full object download, and the request is forwarded to the origin as is.

Note Cisco Support for Microsoft Windows update is enabled by default, and enabled only for specific sites. The enabled sites are updated via OTT metadata.

If you want to disable Cisco Support for Microsoft Windows Update, you must disable OTT caching. To do this, uncheck the Over the Top Cache check box. However, note that unchecking the Over the Top Cache check box disables all OTT functionality, both global and custom OTT configurations.

Cisco WAAS CLI Commands Used with Akamai Connect

This section contains the following topics:

Cisco WAAS Global Configuration Commands Used with Akamai Connect

The following table highlights the Cisco WAAS global configuration commands used with Akamai Connect.

Table 12: Cisco WAAS Global Configuration Commands Used with Akamai Connect

Command	Description
(config) accelerator http object-cache connected enable	Enables the Akamai Connected Cache to cache content that is delivered by an Edge server on the Akamai Intelligent Platform. Object caching is done on the client side Cisco WAAS device only. **Note** You must enable and register Akamai Connect from the Cisco WAAS Central Manager before you run the **accelerator http object-cache connected enable** global configuration command. Otherwise, running this command will invalidate the Akamai Connect End-User License Agreement.
(config) accelerator http object-cache cws-check enable	Enables Cisco Cloud Web Security feature.
(config) accelerator http object-cache enable	Turns on the Akamai cache engine for the Cisco WAAS device.
(config) accelerator http object-cache ott enable	Enables Over the Top (OTT) caching.
(config) accelerator http object-cache transparent enable	Enables the Akamai HTTP object cache (the Akamai cache engine) in **Basic** mode. **Note** When using the CLI to enable the HTTP object cache (the Akamai cache engine), the default caching mode is **Basic**. When using the Cisco WAAS Central Manager to enable HTTP object cache, the default caching mode is **Standard**.
(config) accelerator http object-cache transparent advanced	Enables the Akamai HTTP object cache (the Akamai cache engine) in **Advanced** mode.
(config) accelerator http object-cache transparent basic	Enables the Akamai HTTP object cache (the Akamai cache engine) in **Basic** mode.
(config) accelerator http object-cache transparent bypass	Enables the Akamai HTTP object cache (the Akamai cache engine) in **Bypass** mode.
(config) accelerator http object-cache transparent enable	Enables the Akamai HTTP object cache (the Akamai cache engine) in **Basic** mode.
(config) accelerator http object-cache transparent standard	Enables the Akamai HTTP object cache (the Akamai cache engine) in **Standard** mode.

Command	Description
(config) accelerator http object-cache validate-address bypass	Adds bypass server IP addresses to a whitelist for Server Address Validation.
(config) accelerator http object-cache validate-address bypass	Validates the IP server address configuration.
(config) device mode application-accelerator profile branch	For use with Cisco WAVE devices, to enable the device to function as a branch device, to configure resource pre-allocation resources for various Cisco WAAS services to be branch traffic scenario and branch services.
(config) http-object cache validate address enable	Validates the server IP address configuration.

Cisco WAAS Preposition Configuration Commands Used with Akamai Connect

The following table highlights the Cisco WAAS preposition configuration commands used with Akamai Connect.

Table 13: Cisco WAAS Preposition Configuration Commands Used with Akamai Connect

Command	Description
(config-preposition) accelerator http preposition dre enable	Enables Data Redundancy Elimination (DRE) for preposition connections.
(config-preposition) accelerator http preposition task task-name	Configure a preposition task for one or more sites.

Cisco WAAS EXEC Commands Used with Akamai Connect

The following table highlights the Cisco WAAS EXEC commands used with Akamai Connect.

Table 14: Cisco WAAS EXEC Commands Used with Akamai Connect

Command	Description
clear cache http-object-cache invalidate	Clears the HTTP object cache.
clear statistics accelerator http object-cache	Clears HTTP object cache statistics from a Cisco WAAS device.
debug accelerator http object-cache	Enables object cache debugging.
debug cms {router-config | stats}	Monitor and record CMS debugging for router configuration and statistics, from the Cisco WAAS Central Manager.

Command	Description
disk delete-data-partitions	Deletes all data partitions on all logical drives. Data partitions include the CONTENT, PRINTSPOOL, and GUEST partitions. These partitions include all DRE cache files and print spool files.
show accelerator http object-cache	Displays HTTP object cache configuration and status information for a Cisco WAAS device.
show hosts	Lists and verifies the name servers and their corresponding IP addresses, and lists the hostnames, their corresponding IP addresses, and their corresponding aliases (if applicable).
show statistics accelerator http preposition	Displays preposition task information for a Cisco WAAS device.
show statistics accelerator http object-cache	Displays object cache statistics for a Cisco WAAS device.